AD-A164 596

AFGL-TR-85-0130

ENVIRONMENTAL RESEARCH PAPERS, NO. 921

Factors Affecting the Vertical Motion of a Zero-Pressure, Polyethylene, Free Balloon

JAMES F. DWYER

31 May 1985

AEROSPACE INSTRUMENTATION DIVISION PROJECT 7659

AIR FORCE GEOPHYSICS LABORATORY

HANSCOM AFB, MA 01731

BEST AVAILABLE COPY

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

ADA 164596

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION: Unclassified

1b. RESTRICTIVE MARKINGS: NA

2a. SECURITY CLASSIFICATION AUTHORITY:

2b. DECLASSIFICATION/DOWNGRADING SCHEDULE:

3. DISTRIBUTION/AVAILABILITY OF REPORT: Approved for public release; distribution unlimited.

4. PERFORMING ORGANIZATION REPORT NUMBER(S): AFGL-TR-85-0130 ERP, No. 921

5. MONITORING ORGANIZATION REPORT NUMBER(S):

6a. NAME OF PERFORMING ORGANIZATION: Air Force Geophysics Laboratory

6b. OFFICE SYMBOL (If applicable): LCA

7a. NAME OF MONITORING ORGANIZATION:

6c. ADDRESS (City, State and ZIP Code): Hanscom AFB Massachusetts 01731

7b. ADDRESS (City, State and ZIP Code):

8a. NAME OF FUNDING/SPONSORING ORGANIZATION:

8b. OFFICE SYMBOL (If applicable):

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER:

8c. ADDRESS (City, State and ZIP Code):

10. SOURCE OF FUNDING NOS.

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT NO.
62101F	7659	765911	76591114

11. TITLE (Include Security Classification): Factors Affecting the Vertical Motion of a Zero-Pressure, Polyethylene, Free Balloon

12. PERSONAL AUTHOR(S): Dwyer, James F.

13a. TYPE OF REPORT: Scientific Interim

13b. TIME COVERED: FROM 12/31/83 TO 12/31/84

14. DATE OF REPORT (Yr., Mo., Day): 1985 May 31

15. PAGE COUNT: 106

16. SUPPLEMENTARY NOTATION:

17. COSATI CODES

FIELD	GROUP	SUB. GR.

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number): Balloons, Drag Coefficient, Polyethylene, Reynolds Number, Ascent Trajectory Model, Froude Number (over)

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

This paper critiques existing aerodynamic-thermodynamic models for predicting the vertical motion of free balloon systems. It demonstrates that: (a) the aerodynamic drag coefficient model should be based on Froude number and fractional volume as well as Reynolds number, (b) there has been a widespread error in definition of the instantaneous mass of the balloon film involved in the heat transfer process; (c) the gas bubble cannot be modelled as a sphere; (d) the gas bubble is asymmetrical except when near or at its natural ceiling altitude; and (e) the actual gas bubble shape, and most probably the added mass, is directly related to the type of gore pattern. Finally, a procedure is proposed for the analysis of actual flight data to enable the development of a practical, but also theoretically sound, model of the aerodynamic drag coefficient of a zero-pressure, free balloon – and subsequent refinement of the heat transfer models for direct and reflected solar energy.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT: UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☐ DTIC USERS ☐

21. ABSTRACT SECURITY CLASSIFICATION: Unclassified

22a. NAME OF RESPONSIBLE INDIVIDUAL: James F. Dwyer

22b. TELEPHONE NUMBER (Include Area Code): (617) 831-3005

22c. OFFICE SYMBOL: LCA

DD FORM 1473, 83 APR EDITION OF 1 JAN 73 IS OBSOLETE.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

18. SUBJECT TERMS (Contd)

Heat Transfer Model

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

Preface

Equally as important as balloon thermodynamic problems (and perhaps more challenging) are balloon aerodynamic drag problems, and the interrelated problems of dynamically determined balloon bubble shapes. A review of the literature reveals that far from being solved, these latter problems have barely been defined. Although we can draw no lasting comfort from knowing that our status is not unique, we should certainly be encouraged by the fact that others are vigorously (and with some success) pursuing solutions to very similar problems. A quite concise statement of our mutual difficulty is the following:[1]

> The most basic problem of determining the equilibrium figure of (the body) also requires a simultaneous evaluation of the flow around (the body), which in turn depends on the shape. It is doubtful that (the body) attains a true equilibrium figure under natural conditions, and the analytical problems arising from the coupling of the flow and shape are unsolved, even for the steady case.*

Quite simply stated, this paper re-evaluates the efforts that have gone into producing practical aerodynamic-thermodynamic, flight performance models. It has been justified primarily by those programs that require accurate ascent rate prediction or the ability to account for the effects of balloon motion, insofar as

---

* In the quotation we have substituted the term "the body" wherever Green used the term the raindrop.

1. Green, A. W. (1975) An approximation for the shapes of large raindrops, J. Appl. Meteorol. 14:1578-1583.

they affect data obtained from balloon-borne sensors. Although our findings are significant, they are not conclusive. However, it is hoped that the results will be useful from both theoretical and practical perspectives. To the extent that they are, much of the credit must go to Mrs. Catherine Rice, who is both the work unit and task scientist, without whose con-tructive criticism the arguments herein would not have been as well organized or (we trust) as convincing.

Contents

Contents

Illustrations

Tables

Factors Affecting the Vertical Motion of a Zero-Pressure, Polyethylene, Free Balloon

1. INTRODUCTION

1.1 Objectives

Knowledge of the vertical motions of free balloon systems is important for four reasons. First, to effect a desired ascent profile, we need to understand how a balloon system responds, both to its free-lift force and to changes in its operational environment. Second, to initiate ascent (or descent) or to vary the rate, we must be able to predict the effects of both deballasting and gas valving. Thirdly, to design a balloon-borne experiment, we often need to be able to predict balloon system motions. Fourth, to reduce certain flight sensor data, we must be able to account for effects of system motions on scientific observations. With these needs clearly established, we re-present the relevant [illegible], and restate the problems with historical comments and fresh insights. In this way, we seek to bring about better understanding of the problems, greater ease in modifying them, and improved agreement between observed and predicted system performance.

(Received for Publication 30 May 1985)

1.2 Background

The first step toward achieving our objectives was to review the rather extensive literature in two of three problem areas that deal with balloon ascent and float motions. These first two areas and the more significant works reported thereunder are:

(a) Ascent Rate Prediction Models
- 1941 Clarke and Korff[2]
- 1952 University of Minnesota[3]
- 1955 Ney and Winkler[4]
- 1958 Erickson and Froehlich[5]
- 1974 Nelson[6]
- 1976 Kremser[7]

(b) Comprehensive Flight Performance Models
- 1949 Smith and Murray[8]
- 1952 Hall[9]
- 1952 University of Minnesota[10]
- 1961 Emslie[11]
- 1963 Dingwell et al[12]
- 1966 Germeles[13]
- 1970 Hansen[14]
- 1973 Fujii et al[15]
- 1974 Kreith and Kreider[16]
- 1978 Romero et al[17]
- 1978 Balis Crema[18]
- 1981 Carlson and Horn[19]
- 1981 Carlson and Horn[20]

The third problem area, Float Altitude Motions, encompasses some of the principal concerns that compelled this study. Because of the scale of these motions, and the strictly operational nature of most free balloon efforts, it is not surprising that literature in this ar a is scarce. However, there is extensive reporting on related natural motions of the atmosphere, and on the motions of floating superpressur: balloons. A chronological sampling of such works includes:

Because of the large number of references cited above, they will not be listed here. See References, page 41.

(c) Float Altitude Motions

1949 Smith and Murray[8]

1950 Emmons et al[21]

1952 University of Minnesota[22]

1966 Hirsch and Booker[23]

1968 Nishimura and Hirosawa[24]

1969 Morris and Stefan[25]

1971 Nishimura et al[26]

1974 Levanon[27]

1976 Levanon and Kushnir[28]

1977 Julian et al[29]

1978 Massman[30]

We must keep in mind that in the late 1940s there was no one who had any experience with large balloons made from inextensible plastic films. In truth, balloons having volumes in excess of 10 million ft^3 did not become common until 1959, and early balloon envelopes were made from either relatively inextensible coated fabrics or the very extensible radiosonde balloon material. We must also bear in mind that a large-scale digital computer was seldom available in the early days of the plastic balloon; this both prevented and discouraged attempts to solve or even to define rigorously many of the problems.

---

21. Emmons, G., et al (1950) Oscillations in the stratosphere and high troposphere. Bull. Am. Meteorol. Soc. 31(No. 4):135-138.

22. University of Minnesota (1952) Progress Report on Research and Development in the Field of High Altitude Balloons, Volume IX, Contract Nonr-710(01).

23. Hirsch, J. H., and Booker, D. R. (1966) Response of superpressure balloons to vertical air motions, J. Appl. Meteorol. 5:226-229.

24. Nishimura, J., and Hirosawa, H. (1968) The hunting mechanism of plastic balloons, ISAS Bull. 4(1B):93-110.

25. Morris, A. L., and Stefan, K. H. (1969) High Altitude Balloons as Scientific Platforms, National Center for Atmospheric Research.

26. Nishimura, J., et al (1971) Balloon behavior during level flight, ISAS Bull. 7(1C):257-268.

27. Levanon, N., et al (1974) On the behavior of superpressure balloons at 150 mB, J. Appl. Meteorol. 13:494-504.

28. Levanon, N., and Kushnir, Y. (1976) On the response of superpressure balloons to displacements from equilibrium density level, J. Appl. Meteorol. 15:346-349.

29. Julian, P., et al (1977) The TWERLE experiment, Bull. Am. Meteorol. Soc. 58(No. 9):936-948.

30. Massman, W. J. (1978) On the nature of vertical oscillations of constant volume balloons, J. Appl. Meteorol. 17:1351-1356.

1.2.1 ASCENT RATE PREDICTION MODELS

Early balloon users sought (and we still seek) mathematically simple models for predicting balloon ascent rate. Even now we anticipate that a comprehensive aerodynamic-thermodynamic flight performance model will enable us to perform the types of factor sensitivity analyses necessary for the development of such a rate prediction model. However, with the increased capabilities of new microcomputers we may soon have both the capacity and the speed needed to solve the comprehensive flight performance model (both interactively and in real-time) on location at remote balloon launch sites.

The University of Minnesota[3] developed the most frequently used and studied ascent rate prediction model. Gildenberg* used this model to analyze many balloon ascents; his objective was to improve the accuracy and applicability of the model by refining its thermodynamic and aerodynamic coefficients. Nelson's efforts were directed along the same lines.[6] He published the results of these efforts to use the model for: (a) balloons with volumes from 1 to 30 million ft^3, (b) balloons carrying payloads weighing between 100 and 10,000 lbs, and (c) balloons ascending before and after sunset. Like Gildenberg, he had limited success.

Considering what might be learned from using a complete flight performance model (such as proposed herein), it is possible that the University of Minnesota's model might be enhanced to serve as a practical ascent rate predictor. Thus, we reproduce the model so that the reader might appreciate its relative simplicity (compared to the proposed comprehensive flight performance model). We note that the originators urged caution in its use.†

```
F = C1 * (G * [L * v + 4 * v]**3/[T**2])**(0.25)
     + C2 * (v**2) * (P * G/T)**(1/3)
```

where

F is the free lift force, normalized by dividing by the weight of the displaced air,

C1 is the thermodynamic coefficient, 7.4E-04,

---

*B. D. Gildenberg retired as a meteorologist and balloon operations controller at the AFGL's Holloman AFB balloon facility in New Mexico. His work on this problem is contained in unpublished notes and letters to his coworkers.

†Arithmetic operations in this report are expressed in FORTRAN operator symbols. These symbols are widely recognized, but we include their definitions here to ensure against misinterpretation: addition (+), subtraction (-), multiplication (*), division (/), and exponentiation (**).

G is the weight of the displaced air, lbs,

L is the atmospheric lapse rate, degrees C per 1000 ft,

v is the ascent rate, ft per min,

T is the air temperature, degrees K,

C2 is the aerodynamic coefficient 6.5E-07,

P is the atmospheric pressure, mB.

Kremser's ascent rate prediction model[7] seems to be the only model that is at least as sophisticated as that of the University of Minnesota, but it has not been extensively evaluated. Therefore, we believe that a continued effort with this type of model would be totally unproductive at this time.

1.2.2 COMPREHENSIVE FLIGHT PERFORMANCE MODELS

As with the ascent prediction models, we find that there are two main problem areas, namely, aerodynamic drag and thermodynamic drag, now designated as heat transfer. Smith and Murray[8] treated both, but provided limited details on flight thermodynamics. Their primary thermodynamic concerns were atmospheric temperature lapse rate and the balloon gas superheat.

They treated aerodynamic drag in the conventional manner; they used a spherical shape – the only comparable shape for which drag data was readily available. On the other hand, they did note that, during the early part of the ascent, balloon shape is characterized by a "flabby, unfilled portion", which would affect the aerodynamic drag. Although this was a significant observation, they failed to capitalize on it because, perhaps, balloon float altitudes at that time were relatively low. Thus, the flabby portion would exist for an insignificant time during the ascent.

Smith and Murray made a further observation. They noted the inability to solve the general equations of motion using methods then available.

Hall[9] recognized aerodynamic and thermodynamic drag problems as two of the principal difficulties of his day:

> The greatest uncertainties in the analytical premises are in the magnitude and variation in the drag coefficient, ... and the mechanism and rate of heat transfer between the balloon and its environment.

Apparently for the same reasons as Smith and Murray, he used the sphere as his model for balloon shape. However, he noted that the balloon's "flexibility" would "certainly" be a governing factor with respect to the drag coefficient.

Hall appears to have been the first to include in a flight performance model the heat transfer processes: (a) free and forced convection between air and balloon film, (b) free convection between inflation gas and balloon film, (c) solar energy input, and (d) infrared heat exchange between balloon film and environment.

His work was organized for analysis rather than for solution by numerical methods; his style is that of University of Minnesota researchers.

One cannot overemphasize the efforts and contributions of the University of Minnesota research team to both the aerodynamics and thermodynamics of free-flight plastic balloons. Their work, which also included design of both balloons and flight instrumentation, is recorded in 16 volumes (with flight data). In these volumes there are four specific comments[3, 10] on aerodynamic drag problems that are most relevant to our effort to provide an improved, reformatted, comprehensive, flight performance model:

> Unfortunately in the case of an ascending nonextensible type balloon, the shape is not constant with altitude; varying from the shape of a small sphere with long depending folds of fabric at take-off to roughly a spherical shape at altitude. It is not possible therefore to use a single function CD(Rn) to predict the drag at all altitudes.
>
> The difficulty in predicting the value of CD in advance lies in the fact that the shape of the balloon is not constant with altitude, and thus one cannot carry over the results of wind-tunnel experiments on any particular shape of model.
>
> The described dynamic pressure loading on the balloon is such as to make the balloon more oblate than its original natural shape.
>
> It is not difficu... to imagine that a balloon free to change shape with dynamic forces will hav... lone so appreciably before the velocity of dimpling has been reac...ed.*

Again, the difficulty in solving the drag problem seems to have caused investigators to ignore it ... at least to the extent that they did not attempt to synthesize their observations into a formal statement ... he problem.†

Emslie[11] was the first person to p... an aerodynamic-thermodynamic flight performance model in a system ... ations for solution on a large-... le digital computer. This system of ec... s was his means of investigating ...alloon dynamics. It included: (a... perfec... gas law, (b) an equation of vertical motion, (c) a gas energy equation and (d) ... equation for fabric or film energy (film is now the preferred no... .

Emslie made two e... ...nt points that relate to our current effort. First, although he used a constant drag coefficient (again a technological expedient), he noted the gross asymmetries in the folds of the film below the gas

---

*Dimpling velocity is defined as that velocity at which the dynamic pressure induced by the ascent rate equals the internal pressure on the crown of the balloon.

†"... an...ther characteristic which unifies science is the ability to ignore problems ...h are not yet capable of solution ..."[31]

31. Rive..., P. J. (1983) A world in which nothing ever happens twice, J. ...per. Rsch. Soc. 34(No. 8):681.

bubble when the balloon is below its float altitude (Figure 1). Second, he expressed the mass of the film surrounding the gas bubble as a function of the enclosed volume. Although he understated the mass, basing it on the surface area of a sphere with a volume equal to the volume of the enclosed gas bubble, he nevertheless recognized that the mass of the film involved in the heat transfer process was not constant (see Figure 2).

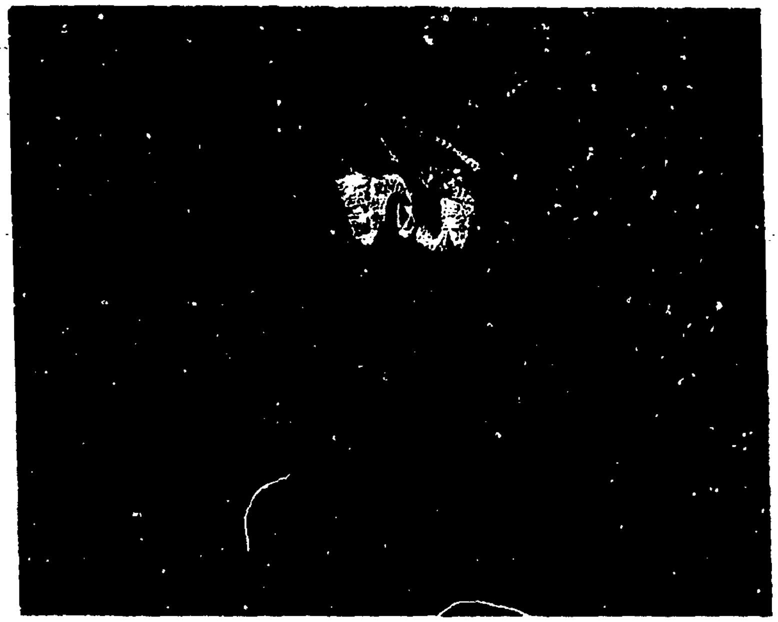

Figure 1. Representative Configuration of an Ascending Tailored Natural Shape Balloon, 2.01 Million ft^3, Flight No. H81-012. Note the extreme asymmetry of both the gas bubble and folds of undeployed film below the bubble shortly after launch

Prior to Emslie's work, performance models were used for relatively low altitude balloon flights and relatively small balloons (with a few exceptions, of course). His work came at a time when we were beginning to fly routinely at altitudes well above 100,000 ft and on balloons with volumes of 10,000,000 ft^3 and larger. For such high performance systems, ascent ballast capacity was at a premium and ascent times were on the order of hours — especially at night. Thus, models developed to forecast accurately the ascent profile required even more accurate formulation. Therefore, it is unfortunate that, in expanding Emslie's work, and in translating it into computer code, Dingwell et al[12] restated the film energy equation, using constant film mass. Instead, they should have redefined Emslie's model of the relationship between shell mass and instantaneous

Figure 2. A Large Tailored Natural Shape Balloon, 21.77 Million ft^3. Note that the balloon envelope material is not concentrated around the gas bubble, but is distributed between the gas bubble and the "rope" of film yet to be deployed. Balloon is shown shortly after launch on Flight No. H78-052

volume, expressing the fact that the actual area of the balloon surface that encloses the gas is greater than the surface area of the enclosing shape, but that it is far less than the constant area of the whole envelope (see Figure 3), except in the vicinity of the natural float altitude.

In continuing Emslie's work, Dingwell et al developed a system of nine simultaneous equations to solve for the following dependent variables as functions of time:

(1) altitude
(2) vertical velocity
(3) gas temperature
(4) film temperature
(5) gas weight
(6) instantaneous balloon volume
(7) atmospheric pressure
(8) atmospheric density
(9) payload weight.

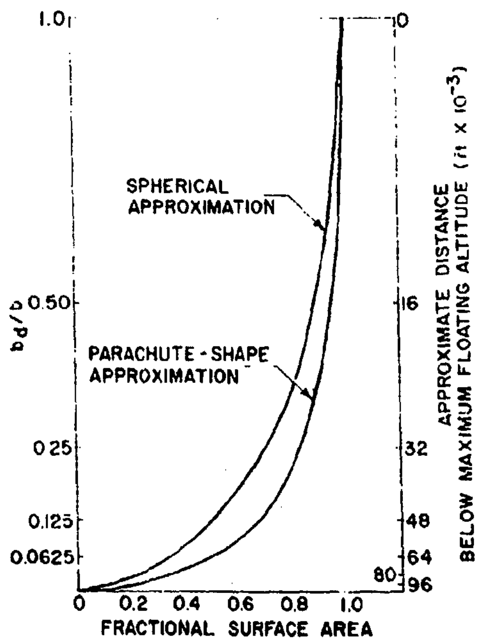

Figure 3. Approximate Mass of the Shell Surrounding the Gas Bubble as a Fraction of Total Film Mass, in Relation to the Ratio of Instantaneous Specific Lift to Specific Lift at Natural Ceiling Altitude. The approximation based on the simplified parachute shape model is given in subroutine MYBLN. The spherical approximation is based on the function:

R = 0.5 * {1 - cos [π* (V/Vmax)**(1/3)] },

where R is the fractional surface area, Vmax is the maximum volume, and V is the instantaneous volume. Note that for a balloon with a ceiling altitude of 100,000 ft, the fractional surface area at launch is about 10 percent of the entire envelope area

To solve these equations, one had to determine certain parameters as functions of time and the above dependent variables. These were the relevant heat transfer coefficients, heat transfer areas, and optical properties. For these they provided tables, graphs, and mathematical models, including those for heat transfer, both by forced and free convection, and by radiation.

Germeles[13] reported extensions and improvements of analyses and computer codes reported by Dingwell et al. Furthermore, he continued the constant film mass error. Hansen[14] made use of this work by Germeles, but apparently did not modify it.

In Japan Fujii et al[15] developed a far less sophisticated routine; it included a constant drag coefficient (0.2) and a he .ansfer model that did not explicitly contain the film mass term.

Kreith and Kreider[16] further refined the work of Germeles and Dingwell et al, but made no reference to Emslie. Although they added a routine to compute CD as a function of Reynolds number, and made significant changes in some of the heat transfer models, they left the computer codes substantially the same as those reported by Germeles. Their work is now the generally accepted standard.

Romero et al[17] referred to the works of both Germeles, and Kreith and Kreider, but Balis Crema et al[18] referred to only Kreith and Kreider. Neither work cited Emslie and both continued to use constant film mass. Romero et al, however, did define separate values for drag coefficients in turbulent and laminar flows, 0.45 and 1.35 respectively.

Carlson and Horn[19] followed the lead of Kreith and Kreider, using a system of eight roughly "equivalent" equations — including the film mass term, still represented as a constant. However, they modified the assumptions to allow the inflatant to absorb and emit energy. This is a significant change that requires further study before general acceptance, especially in light of some of the changes in this proposed model.

With regard to balloon shape, Carlson and Horn assumed that

> ... over much of the flight profile the balloon shape is close to a sphere..

and, accordingly, also used a drag coefficient model based on Reynolds number only. In addition, they observed that computed

> ..balloon ascent velocities between launch and the tropopause are very sensitive to the values of CD,

and suggested that the balloon could

> ...experience significant skin friction drag in addition to the pressure drag normally found on a sphere.

Carlson and Horn[20] added significantly to their previous observations when they commented that the apparent virtual mass coefficient, used in the first equation, might be inappropriate for "the balloon configuration."

1.2.3 FLOAT ALTITUDE MOTIONS

The vertical motions of a zero-pressure balloon at float altitude are complex. If we are to understand them at all, we must also understand the zero-pressure balloon's interaction with its use environment. Although we found that very little exists on this subject, we did find the following to be both pertinent and interesting.

(a) Smith and Murray[8] noted the influence of vertical winds on vertical balloon motions. They even included these wind models in their flight performance model, but the effects at today's relatively "high" altitudes may be insignificant.

(b) Nishimura and Hirosawa[24] treat a "hunting" motion that relates to the balloon construction (a subject area not to be ignored with regard to its effects on balloon performance at natural float altitude).

(c) Massman[30] comments that "... the Brunt-Vaisala oscillations of the balloon's EDS (equilibrium density surface) can have a period as short as 5 min in the stratosphere."*

1.3 Conclusions Based on the Literature Search

Based on findings and observations that appear in the cited literature, we believe that four areas deserve particular attention in the formulation of any flight performance model: (a) balloon shape during ascent, (b) mechanism for heat transfer between the gas and the film, (c) aerodynamic drag coefficient; and (d) added mass.

1.3.1 BALLOON SHAPE DURING ASCENT

The shape taken by the partially full balloon (gas bubble), while either ascending or floating, governs the effective envelope mass, the gas bubble surface area, and the areas involved in heat transfer processes (Figure 4). Also during ascent and descent the shape affects the drag area (Figure 5) and the air flow around the balloon, hence, the aerodynamic drag coefficient. Possibly, as we shall see, it also affects the added mass.

Clearly ascending (or descending) balloon shapes are far from spherical — even though the leading surface of the gas bubbles in Figures 6 and 7 appear to be hemispherical. Factually, a partially full balloon is asymmetrical in every plane, and this asymmetry is further exaggerated by the gore deployment, which is governed in turn by the gore pattern.† Figures 1, 2, and 4 show bubble shapes quite typical of today's large, fully tailored, natural shape balloons; the maximum

---

*The Brunt-Vaisala period is defined to be: 2 * π/SQRT [g * (Beta + dT/dh)/T] seconds, with the terms defined in Brunt.[32]

†For a short commentary on the development of balloon gore patterns see Dwyer.[33]

32. Brunt, D. (1927) The period of simple vertical oscillations in the atmosphere, Quart. J. Roy. Meteorol. Soc. 53:30-32.

33. Dwyer, J. F. (1978) Zero pressure balloon shapes, past, present, and future, Scientific Ballooning (COSPAR), W. Riedler, Ed., Pergamon Press, pp. 9-19.

Figure 4. Balloon Ascent Shape, Flight No. H84-003. The early ascent shape and relative gas bubble surface area of this tailored natural shape balloon, model no. SV-017B, differs considerably from those of the fully tailored natural shape balloons shown in Figures 1 and 2. It is an intermediate size heavyload balloon having a maximum volume of 5.142 million ft^3

horizontal cross section of each of these balloons is far less circular than those of earlier balloons made with either rectangular or semi-tailored gores (compare the bubble shapes in Figures 1 and 7).

During inflation, the balloons in Figures 6 and 7 assumed the characteristic shape shown in Figure 8. Gore pattern types that produce such shapes are shown in Figure 9 as patterns, numbers 2 and 3, and (to a lesser degree) pattern number 4. On the other hand, Figures 10 through 12 are representative of the prelaunch shapes of balloons, such as those shown during ascent in Figures 1, 2, 4, and 5. These latter shapes are characteristic of balloons made with fully tailored gores, pattern number 1, Figure 9.

In addition to the gas bubble proper, we should also consider the shape and effects of the trailing undeployed balloon shell. This is the mass of film that (as we noted) Smith and Murray called the "... flabby unfilled portion..." and researchers at the University of Minnesota described as "... long depending folds of fabric.." Emslie also noted the gross asymmetries in the folds of undeployed material, and Carlson and Horn called attention to the fact that large balloons, in

Figure 5. Ascent Configuration of a Relatively Small, Fully Tailored Balloon, Model No. LTV-019, Having a Maximum Volume of 628,000 ft^3, Flight No. H81-006. The highly assymetric horizontal cross-section is common in a balloon of this size at liftoff

the early stages of ascent (when the existence of this surplus is most evident), have shapes that are ".. significantly different from that of a sphere.. " It is probable that this shape feature plays an important role in determining the drag coefficient, much as in the case of a sphere with a splitter plate.[34]

To better represent overall balloon shape in a computable configuration, we selected the existing balloon shape model shown in Figure 13.[35] This contrived shape has at least two distinct advantages. First, it provides smooth transition from the modeled, partially full state to the full, natural shape state, a most important consideration in the analysis of vertical motions that occur at or near the natural ceiling altitude. Second, it permits reasonably accurate computation of the instantaneous mass of the balloon film involved in the heat transfer process — including load cap film, if such is present.

34. Hoerner, S. F. (1965) Fluid-Dynamic Drag (published by author).

35. Dwyer, J. F. (1980) The Problem: Instantaneously Effecting Controlled Balloon-System Descent from High Altitude, AFGL-TR-80-0277, AD A100255.

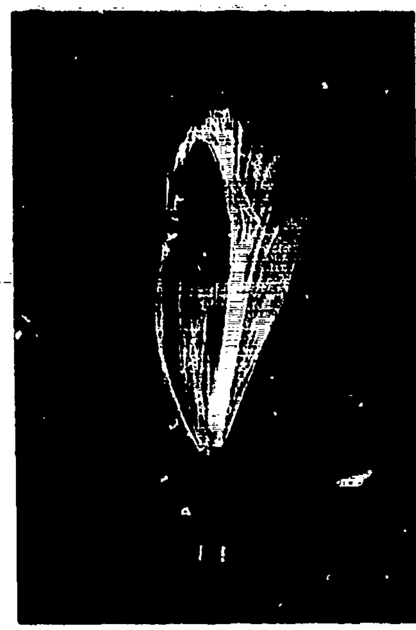

Figure 6. Ascent Configuration of an Early Moby Dick Balloon Having Semi-Tailored or Rectangular Gores and a Maximum Volume Less Than 100,000 ft^3, Flight No. E-149, 27 November, 1953. The hemispherical crown and large air pocket are characteristic of balloons constructed with the aforementioned gore patterns

1.3.2 MECHANISM FOR HEAT TRANSFER BETWEEN THE GAS AND THE FILM

Traditionally, one has assumed that the mechanism for heat transfer between the gas and balloon wall is free convection. We have no direct evidence to support this assumption; neither do we have knowledge of the sensitivity of the models based on this assumed mechanism. However, we do have evidence that an ascending balloon is quite asymmetric (see Figures 1, 2, 5, and 7), and upward looking cameras have shown how balloons rotate considerably during ascent. These two facts suggest that the gas should be in constant motion, agitated by large internal vanes of the envelope material that are surplus to the instantaneous shape. Therefore, one might ask whether this heat transfer is due to forced, rather than to free convection. The differences in computed flight performances based on these opposing assumptions are not known; should they not be significant, then the more easily computed model should be used. We do not yet have sufficient reason to break with convention on this issue and consequently continue to use free convection.

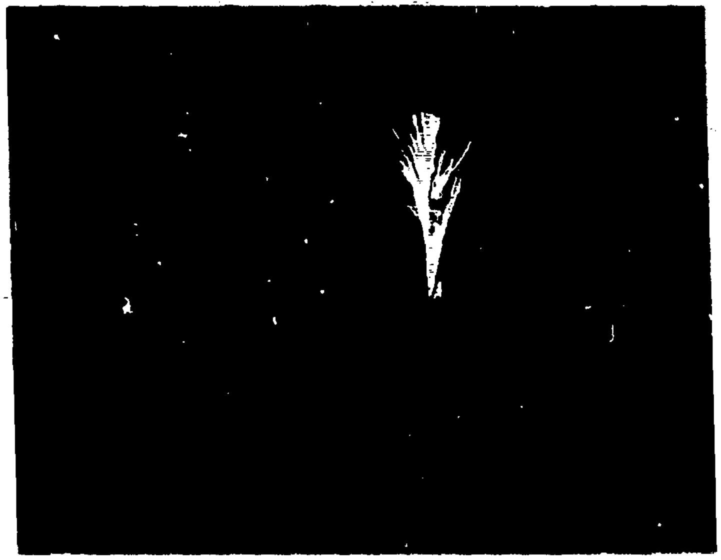

Figure 7. Ascent Configuration of a Semi-Tailored Balloon, Model No. TTV-001F, Flight No. H73-016. This tapeless, semi-cylinder balloon, having a volume of 804,000 ft^3, has the same configuration characteristic of the Moby Dick balloon shown in Figure 6

1.3.3 AERODYNAMIC DRAG COEFFICIENT

Drag coefficient models found in the literature have ranged from a single value,[15] through a five-part, piecewise continuous function of Reynolds number,[16] to the categorical conclusion that a single function relationship between drag coefficient and Reynolds number is not possible.[10] In the literature, three bases for arguments support the latter conclusion: (a) inconstant shape, (b) shape deformability, and (c) dimensional reasoning. Based on the arguments that follow, we conclude that any valid model that is to determine aerodynamic drag coefficients must consider at least two dimensionless variables: Reynolds number and Froude number. Further, because the shape has no single characteristic length, it is reasonable to expect that we will need a third dimensionless variable, fractional volume. This latter variable is defined as the ratio of the instantaneous volume to the maximum volume; it has the effect of normalizing the shape so that a single dimension of the balloon might serve as a characteristic length to determine the effective drag area.

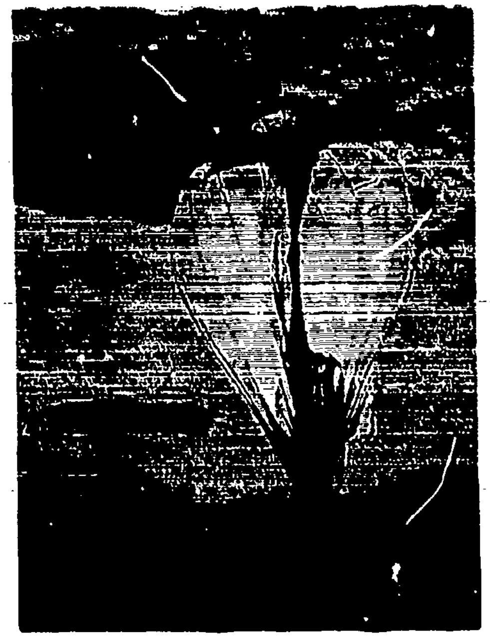

Figure 3. Inflation Configuration of a Semi-Tailored Balloon, Model No. TTV-001F, Flight No. H81-014. During inflation the tapeless, semi-cylinder balloon is characterized by highly irregular deployment of material excess to the amount needed to enclose the gas bubble. What appear to be load tapes are aluminum-backed polyethylene seam reinforcements; these make the system more radar reflective

1.3.3.1 Inconstant Shape

The University of Minnesota study summarizes well the argument based on inconstant shape:

> Unfortunately in the case of an ascending non-extensible type balloon, the shape is not constant with altitude, varying from the shape of a small sphere with long depending folds of fabric at take-off to roughly a spherical shape at altitude. It is not possible therefore to use a single function CD(Rn) to predict the drag at all altitudes.
>
> [emphasis added]

Schlichting,[36] also in this regard, notes that the use of Reynolds number alone presupposes both the same shape and orientation. When one considers the

36. Schlichting, H. (1968) Boundary-Layer Theory, McGraw-Hill Book Co., New York, 6th Edition, p. 16.

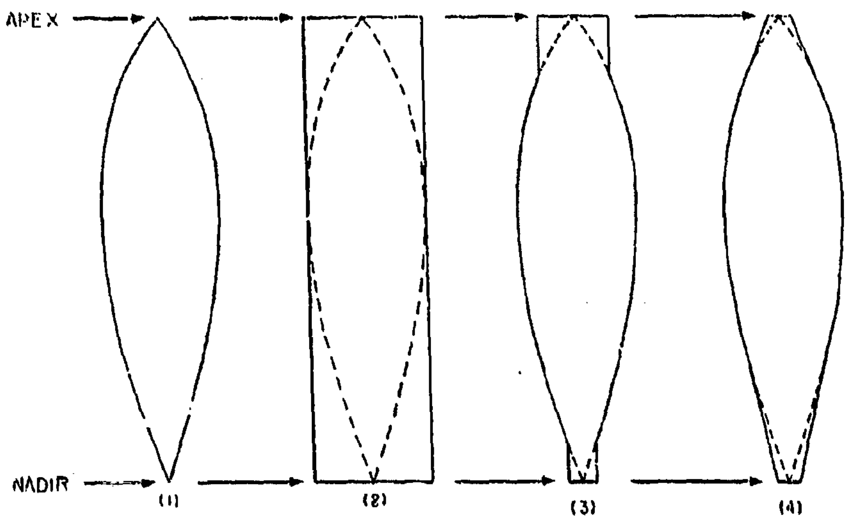

Figure 9. Gore Pattern Types for Natural Shape Balloons. The patterns shown are: (1) fully tailored, (2) rectangular, (3) semi-tailored, and (4) tapered tangent. These respective patterns are used to construct the following balloon types: (a) fully tailored balloons, (b) cylinder balloons, (c) semi-cylinder (sometimes called tailored tapeless) balloons, and (d) tailored balloons

documented balloon shapes and the differences between the forward surfaces presented by rising and descending balloons, it is clear that these two presuppositions are invalid. Hence, use of Reynolds number alone is insufficient.

1.3.3.2 Shape Deformability

We have recognized for a long time that there is a significant difference between the static shapes of free balloons just prior to launch and the dynamic shapes taken by the same balloons during ascent. We concluded from this that: (a) the ascent shapes of a balloon represented deformations of static shapes, (b) the aerodynamic shapes of the balloons were dependent on aerodynamic drag, (c) solutions to the drag problem involved free surface phenomena, and (d) the Froude number should play a role equally as important as the Reynolds number. We are still surprised that the connection between shape deformation, free surface phenomena, Froude number, and aerodynamic drag was not made long ago, in the earlier motion studies.

In the literature on balloons, there are numerous observations that suggest the applicability of free surface phenomena to the aerodynamic drag problem. Hall implied such in his use of the term "flexibility", and in his certainty that flexibility would be a governing factor in the determination of drag. The University of Minnesota study likewise implied the relevance of such phenomena when it

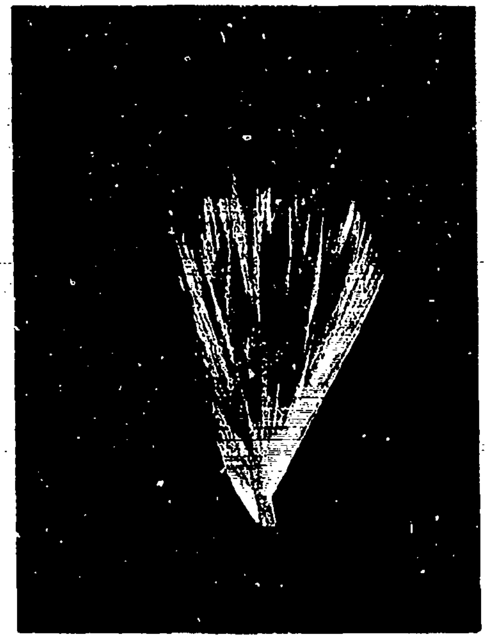

Figure 10. Inflation Configuration of a Fully Tailored Balloon, Model No. LTV-018, Having a Maximum Volume of 355,000 ft^3, Flight No. H80-029. From the top end-fitting outward, the fully tailored gores are uniformly deployed

described the dynamic pressure acting downward on the top of the balloon, making it ". . more oblate than its original natural shape." It [U. of M.] further reinforced our unique and unusual interpretation by the comment, "It is not difficult to imagine that a balloon free to change shape with dynamic forces will have done so appreciably before the velocity of dimpling has been reached." [emphasis added] Interpretation of this deformation as a free surface phenomenon is also strengthened by the Carlson and Horn reference to the possibility that the balloon apex region, during the early stages of ascent, might be ". . . more oblate than a sphere due to pressure differences across the film."

Where such deformations do occur, Schlichting notes that drag based only on Reynolds number is invalid and that the Froude number must be considered. Indeed, when we re-examined the University of Minnesota's treatment of dimpling velocity, we found that the ratio of dynamic pressure to static internal pressure could be reduced to the Froude number.

We considered three other aerodynamic problem areas to be potentially enlightening with respect to free surface phenomena and Froude number, insofar as they relate to shape deformability: they were raindrops, parachutes, and

Figure 11. Inflation Configuration of a Fully Tailored Balloon, Model No. LTV-013A, Having a Maximum Volume of 2.9 Million ft^3, Flight No. H80-039. From the top end-fitting outward, the fully tailored gores are uniformly deployed

air-supported structures. Although none of these areas yielded anything directly applicable to our problem, the review did provide some rewarding insights.

Gillaspy[37] commented on the raindrop problem, one which is quite analogous to ours:

> A sphere falling in a fluid medium will attain a constant or terminal velocity. When falling at terminal velocity, all of the forces on the sphere are in equilibrium. If the sphere is composed of solid material, this equilibrium is the balance between the weight and the aerodynamic drag forces on the sphere. However, in a liquid drop the balance is much more complicated. Other forces arise from the fact that the drop is liquid and deformable. [emphasis added]

37. Gillaspy, P. H. (1981) Experimental Determination of the Effect of Physical Properties on the Drag of Liquid Drops, Ph.D. Dissertation, University of Nevada (funded under U. S. Army Research Office Contract No. DAAB29-77-G-1072).

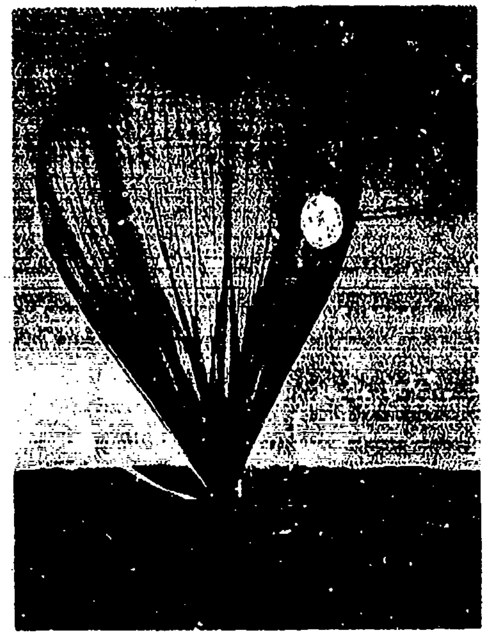

Figure 12. Inflation Configuration of a Fully Tailored Balloon, Model No. SV-017B, Having a Maximum Volume of 5.142 Million ft^3, Flight No. H84-003. From the top end-fitting outward, the fully tailored gores are uniformly deployed

He developed a model that accounted for influences of Reynolds number and Bond number. Like the Froude number in our proposed approach to balloon aerodynamic problems, the Bond number accounts for the effects of gravity.

Perhaps balloon problems are generically closer to parachute problems than they are to raindrop problems, primarily because the stresses in a balloon shell are closer to stresses in a parachute canopy than they are to surface tensions in a raindrop. On this premise, Von Karman's introduction of the Froude number into the analysis of parachute opening shock[38] provides some additional encouragement that the Froude number might indeed be one key to the solution of aerodynamic drag problems involving balloons.

---

38. Von Karman, T. (1945) Note on Analysis of the Opening Shock of Parachutes at Various Altitudes, A. A. F. Scientific Advisory Group.

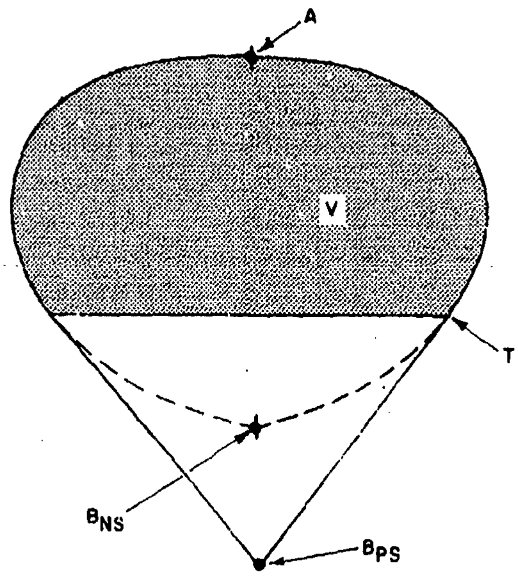

Figure 13. Simplified Parachute-Shape Model. In this model, V is the volume of the inflation gas, A is the apex, B_{NS} is the base of the natural shape generator, B_{PS} is the base of the model, T is the point of tangency between the lower portion of the model and generator shape, and the actual balloon gorelength is the distance between A and B_{PS}

We found in the recent literature on air-supported structures only limited references to Froude number; these were with respect to large tensioned, pneumatic structures.[39,40] We cite them only to indicate that application of free surface phenomena to problems dealing with deformable barrier surfaces has further, and more recent, precedent.

Clearly balloon deformability justifies our interpretation of balloon aerodynamic drag as a free surface problem. Furthermore, it supports the conclusion that we cannot determine aerodynamic drag coefficients by Reynolds number alone; that we must also consider another dimensionless variable — the Froude number.

---

39. Tryggvason, B. V., and Isyumov, N. (1978) Similarity requirements for inflatable structures, Proceedings of the Third U. S. National Conference (on Wind Engineering Research), University of Florida, Gainesville, Florida, pp. 335-338.

40. Tryggvason, B. V. (1979) Aeroelastic modelling of pneumatic and tensioned fabric structures, Proceedings of the Fifth International Conference (on Wind Engineering), Fort Collins, Colorado, pp. 1061-1072.

1.3.3.3 Dimensional Reasoning

Finally, Landau,[41] on the basis of dimensional reasoning argues:

> If the force of gravity has an important effect upon the flow, then the latter (drag force) is determined not by three but by four parameters ..(including the acceleration of gravity)... From these parameters we can construct not one but two independent dimensionless quantities. These can be, for instance, the Reynolds number and the Froude number... (and) ... two flows will be similar only if both these numbers have the same values.

1.3.4 ADDED MASS

The conditions under which the added mass term applies are not well defined, and Carlson and Horn[20] questioned (as did we) whether the term is applied properly to the configuration (of the partially full balloon in vertical motion). We assume that their doubt (like ours) applies only under the stated conditions, when that film, excess to the instantaneous bubble shape, can deploy asymmetrically to form pockets of ambient air (see Figures 6 and 7). This type of deployment would have the effect of increasing the volume in the added mass product represented as CM*BUOY [see Eq. (1) in Section 2.1]. This seems quite probable only at low and intermediate altitudes, before the nadir cone angle becomes great enough to prevent the formation of such air pockets. If adjustment of the added mass term is ever deemed necessary, the added mass coefficient CM can probably be developed as a function of fractional volume for each specific balloon construction type. This is not thought to be required for the large, fully tailored balloons that predominate today. Consequently, such a function will not be introduced at this stage in the development of a comprehensive flight performance model.

2. A NEW FLIGHT PERFORMANCE MODEL

In the following model, the operator D[ ] represents the first derivative with respect to time. The definitions of terms in the equations are given once in the subsection on each equation, and again in the glossary, where (along with their dimensions) their FORTRAN names are given. Complex model components are explained separately; otherwise they are clarified by explicit comments included in the program FORTRAN codes or implicitly by references to specific equations and/or figures appearing in cited documents.

---

41. Landau, L. D., and Lifshitz, E. M. (1959) Fluid Mechanics, Addison-Wesley Publishing Co., Inc., p. 63.

2.1 The Differential Equations

The following eight differential equations define the model when the balloon is partially full; either when it is floating or when it is moving vertically, upward or downward. In these cases: (a) the gas volume V is free to expand or contract, (b) the gas pressure Pg is assumed to follow the ambient pressure, (c) the temperature of the gas Tg reacts to the gas expansion or contraction, and (d) the gas weight Wg is constant if the apex valve is closed.

$$D[v] = (BUOY - DRAG - WS)/[(WS + CM * BUOY)/G] \tag{1}$$

$$D[W] = - DB \tag{2}$$

$$D[Z] = v \tag{3}$$

$$D[Tf] = (Q1 + Q2 + Q3 + Q4 - Q5 + Q6 + Q7)/(CF * WE) \tag{4}$$

$$D[Wg] = - VV * SWG \tag{5}$$

$$D[Tg] = (- Q1 - Q7 + Q8 + Q9 - Q10 + Q11 - SW * V * v)/[Wg * (CV + RG)] \tag{6}$$

$$D[V] = V * \{D[Wg]/Wg + D[Tg]/Tg + v/(RA * Ta)\} \tag{7}$$

$$D[Pg] = - SW * v \tag{8}$$

When, however, the volume of the gas bubble equals the maximum volume of the balloon, and the gas in the balloon is still expanding, either gas must be expelled through the ducts or the balloon will eventually burst. To model this venting, new relationships are required; Eqs. (5a) through (8a) [which replace Eqs. (5) through (8)] provide just such a model. In execution: (a) the gas pressure changes as the balloon vents gas, (b) the bubble volume remains constant, (c) the gas temperature reacts to restricted expansion, and (d) the weight of the gas is reduced due to the venting process. Equations for this alternate balloon state are:

$$D[Wg] = - (VV + VD) * SWG \tag{5a}$$

$$D[Tg] = (- Q1 - Q7 + Q8 + Q9 - Q10 + Q11 - Pg * D[Wg]/SWG)/(Wg * CV) \tag{6a}$$

$$D[V] = 0. \tag{7a}$$

$$D[Pg] = Pg * (D[Wg]/Wg + D[Tg]/Tg) . \tag{8a}$$

We introduced a simple version of this duct venting model in a paper on balloon design,[42] and modified it herein to work with the dynamic case. It is a major change from all previous flight performance models. Those models used a routine called burping; the name (origin unknown) is somewhat inelegant, but the routine is mathematically effective.

However, for the analysis of vertical motions of full balloons at float, it is important that performance model outputs represent (as nearly as possible) actual flight performance; use of the burping model precludes this. Yet, even our more sophisticated process does not account for volume increases when balloons change shape due to venting backpressure. For a venting balloon carrying a payload less than or equal to its design payload, the balloon shell is fully deployed and taut; the volume cannot change perceptibly. The same balloon, carrying a significantly heavier-than-design payload, has excess envelope material and excess potential volume when it begins to vent. As backpressure is created and rises due to venting excess gas, gas expands into the potential volume and observed performance may differ considerably from the model's output. The described performance is typical of fully tailored, natural shape, free balloons. Cylinder balloon performance, on the other hand, is considerably different and more complex, but cylinder balloons are no longer used routinely. (See Nishimura and Hirosawa.[24])

2.1.1 THE EQUATION OF MOTION

D[v] = (BUOY - DRAG - WS)/[(WS + CM * BUOY)/G] ,

where

v is the vertical velocity of the system,
BUOY is the weight of the displaced ambient air,
DRAG is the aerodynamic force resisting balloon vertical motion,
WS is the system weight, including the gas,
CM is the coefficient of added or virtual mass,
G is the gravitational constant.

This equation is essentially equivalent to Eq. (1) of Horn and Carlson.[43] However, we have adjusted it to enable the use of weight in place of mass as a primary dimension. It contains references to two previously discussed problem

---

42. Dwyer, J. F. (1982) Polyethylene Free Balloon Design From the Perspective of User and Designer, AFGL-TR-82-0350, AD A127553.

43. Horn, W. J., and Carlson, L. A. (1983) THERMTRAJ: A Fortran Program to Compute the Trajectory and Gas Film Temperatures of Zero Pressure Balloons, NASA Contractor Report 168342.

areas: added mass, expressed by the term CM * BUOY, and aerodynamic drag, expressed by the term DRAG. As already stated, we continue with the traditional approach to added mass, but not, however, to aerodynamic drag. The term DRAG can be expanded as:

DRAG = 0.5 * RO * CD * HC * v * ABS(v) ,

where

RO is the density of the ambient air computed by the subroutine VIRON,

CD is the aerodynamic drag coefficient,

HC is the horizontal gas bubble cross section, as defined in subroutine MYBLN.

We reserve treatment of the term CD to Section 3, where we will provide specific comments on the problems of modeling it.

2.1.2 DEBALLASTING EQUATION

The deballasting equation accounts for reductions in the dead weight payload on the balloon. It is included as a differential equation (rather than a simple function of time) as a matter of choice, and because a prior version of this performance model included provisions for a cryogenic ballast system that will be reinstituted when sufficient theoretical or practical interest arises.

D[W] = -DB ,

where W is the weight suspended beneath the balloon and DB is the ballast pouring rate.

2.1.3 ALTITUDE EQUATION

This is identical to Eq. (2) of Horn and Carlson.[43]

D[Z] = v ,

where Z is the altitude (above msl) of the system.

2.1.4 FILM TEMPERATURE MODEL

D[Tf] = (Q1 + Q2 + Q3 + Q4 - Q5 + Q6 + Q7)/(CF * WE) ,

where

Tf is the balloon film temperature,

Q1 is the rate of free convective heat transfer between the gas and balloon wall,

Q2 is the rate of direct solar energy absorption by the balloon wall,

Q3 is the rate of IR energy absorption by the balloon wall,

Q4 is the rate of free (or forced) convective heat transfer between the balloon wall and the air,

Q5 is the rate of IR energy emission by the balloon wall,

Q6 is the rate of absorption of reflected solar energy,

Q7 is the rate of radiative exchange between the gas and balloon wall,

CF is the specific heat of the balloon film,

WE is the gas bubble envelope weight computed from the balloon shape parameters as determined by subroutines MYBLN and NELSON.

This temperature model and Eq. (4) of Horn and Carlson[43] are comparable, but a significant difference exists between our term WE and their term MASSF. The latter term is a constant; it accounts for the mass of the entire balloon envelope. On the other hand, the term WE refers to only part of the fabricated balloon envelope: that part which, at a given instant, surrounds the bas bubble. WE depends upon both the balloon envelope construction and the degree of inflation. Conceptually, WE was developed independently of, and without recourse to, the work of Emslie. However, it can be viewed and should be viewed, as a more accurate version of his model, even though it is a unique development.

As a factorable term in the denominator, WE (like MASSF) significantly affects the values computed by this function, most particularly for very large balloons and for small fractional volumes. One should also note that we have dropped the product (Tf * D[WE]/WE), a second-order term that results from the fact that WE is not constant.

Terms Q1 through Q7 are treated in more detail in Section 2.2, the section on heat transfer models.

2.1.5 GAS WEIGHT MODEL (SLACK BALLOON)

D[Wg] = -VV * SWG ,

where

Wg is the balloon gas weight,

VV is the apex gas valve discharge rate,

SWG is the specific weight of the balloon gas.

This gas weight model is comparable with Eq. (3) of Horn and Carlson,[43] but it incorporates a mathematical model of the EV-13 apex valve discharge rates based on data in an earlier report.[35]

2.1.6 GAS TEMPERATURE MODEL (SLACK BALLOON)

$$D[Tg] = (-Q1 - Q7 + Q8 + Q9 - Q10 + Q11 - SW * V * v)/[Wg * (CV + RG)] ,$$

where

Q8 is the rate of absorption of direct solar energy by the gas,

Q9 is the rate of absorption of reflected solar energy by the gas,

Q10 is the rate of emission of IR energy by the gas,

Q11 is the rate of absorption of IR energy by the gas,

SW is the specific weight of the ambient air computed by subroutine VIRON,

V is the instantaneous volume of the gas bubble,

CV is the specific heat of the gas (at constant volume),

RG is the specific gas constant for the inflatant.

This gas temperature model is comparable with Eq. (5) of Horn and Carlson[43] and with Kreith and Kreider.[16] Again, terms Q1 and Q7 through Q11 are treated in Section 2.2.

2.1.7 GAS VOLUME MODEL (SLACK BALLOON)

$$D[V] = V * \{D[Wg]/Wg + D[Tg]/Tg + v/(RA * Ta)\} ,$$

where

RA is the specific gas constant for air,

Ta is the ambient air temperature, computed by subroutine VIRON.

The gas volume model is included as a differential equation, rather than as a definite function of temperature, pressure, and mass, for previously cited reasons.

2.1.8 BALLOON GAS PRESSURE MODEL (SLACK BALLOON)

The nature of our duct venting model requires that we include balloon gas pressure as a differential equation for the case of the full balloon; the differential equation is included in the slack balloon case only because it is required by the symmetry of the solution process.

$$D[Pg] = -SW * v ,$$

where Pg is the balloon gas pressure.

2.1.9 GAS WEIGHT MODEL (FULL BALLOON)

This gas weight model differs from Eq. (6) for slack balloons in that it includes the duct venting model previously discussed in Section 2.1. In this regard it differs also from Eq. (3) of Horn and Carlson.[43]

$D[Wg] = -(VV + VD) * SWG$,

where VD is the duct discharge rate.

2.1.10 GAS TEMPERATURE MODEL (FULL BALLOON)

$D[Tg] = (-Q1 - Q7 + Q8 + Q9 - Q10 + Q11 - Pg * D[Wg]/SWG)/(Wg * CV)$.

This gas temperature model is comparable to Eq. (5) of Horn and Carlson[43] and with Kreith and Kreider,[16] except that it has been modified to reflect the effect of venting excess gas. Again, terms Q1 and Q7 through Q11 are treated in Section 2.2.

2.1.11 GAS VOLUME MODEL (FULL BALLOON)

The gas volume model is included as a differential equation, rather than as a definite function of temperature, pressure, and mass, for previously cited reasons.

$D[V] = 0$.

2.1.12 BALLOON GAS PRESSURE MODEL (FULL BALLOON)

Balloon gas pressure is computed as an essential factor in the model of ducts venting excess lifting gas when the volume of the inflatant tends to exceed the maximum volume of the balloon.

$D[Pg] = Pg * (D[Wg]/Wg + D[Tg]/Tg)$.

2.2 Heat Transfer Models

Heat energy added to or lost by the inflatant is a major factor in overall system performance. Added heat energy decreases the density of the inflatant, thereby increasing the buoyancy. When heat energy is lost the effect is opposite. To reiterate, heat transfer models (relative to polyethylene balloon flight analysis) were introduced by Hall[9] prior to the development of flight performance models for solution by large scale digital computers. Over the last 32 years these models have evolved into 11 elements, the last five of which result from efforts by Carlson[44] to correlate theoretical and actual flight performance. Assumptions underlying these last five elements must still be validated, for he notes that the absorption and emission values deduced for the inflatant are not independent of the balloon envelope materials used on the flights from which the data were obtained.

---

44. Carlson, L. A. (1979) A new thermal analysis model for high altitude balloons, Proceedings, Tenth AFGL Scientific Balloon Symposium, Catherine L. Rice, Ed., pp. 187-206.

Given our use of FORTRAN notation and nonsubscripted symbols, and in order eep our nomenclature and symbols reasonably close to those of Carlson and Horn, we express convective heat transfer between a gas and some object as follows:

Q = CH * dT * Area = [k * Nu/L] * dT * Area ,

where

k is the thermal conductivity of the gas,

Nu is a Nusselt number,

L is a length, characteristic of the object's shape (in this application, the maximum horizontal diameter of the assumed shape),

dT is the difference in their temperatures,

Area is the involved surface area of the object,

CH is the convective heat transfer coefficient, generally correlated with the Nusselt number by the relationship: [k * Nu/L].

2.2.1 FREE CONVECTIVE HEAT TRANSFER BETWEEN THE GAS AND THE BALLOON WALL

Q1 = CQ1 * CG * GN1 * Nu * (Tg - Tf) * SA/DM ,

where

CQ1 is a correction coefficient to be established on the basis of experience (initially, CQ1 = 1),

CG is the gas thermal conductivity coefficient,

GN1 is the Nusselt number correction,

Nu is the Nusselt number,

SA is the surface area of the gas bubble shape computed by subroutine MYBLN,

DM is the diameter of the bubble model computed by subroutine MYBLN.

Since we still consider this process as free, rather than forced convection, we represent the Nusselt number in the general form:

Nu = a * [b + c * (Pr * Gr)**(d)] ,

where

a is an arbitrary constant (as are b, c, and d),

Pr is the Prandtl number for Helium, 0.67,

Gr is the Grashof number expressed as G * [(DM * SWG/VSG)**2] * ABS(Tf - Tg)/Tg ,

SWG is the specific weight of the gas,

VSG is the viscosity of the gas,

ABS( ) is the symbol for absolute value.

The values used for the constants a, b, c, and d have not been consistent during the evolution of the flight performance models (see Table 1). We use Carlson's values in this proposed model, except that we include the value of "a" in "GN1", a correction factor. We believe that our recommended changes will require that other empirical constants be changed also, but only after there has been an opportunity to correlate the results with actual flight data.

Table 1. Nusselt Number Model Constants for Free Heat Transfer Between the Gas and the Balloon Film, Showing Similarities and Differences, and Arranged Chronologically. Notes reflect valid ranges for: (1) 10E9 ≤ (Pr * Gr) ≤ 10E12, (2) (Pr * Gr) ≤ 10E9, (3) 10E9 ≤ (Pr * Gr), (4) (Pr * Gr) ≤ 1.5 * 10E8, and (5) 1.5 * 10E8 ≤ (Pr * Gr)

	a	b	c	d	Notes
Germeles[13]	1	0	0.13	0.33	1
Fujii et al[15]	1	0	0.65	0.25	
Kreith and Kreider[16]	3	0	0.59	0.25	2
	3	0	0.13	0.33	3
Balis Crema et al[18]	1	0	0.12	0.33	
Carlson and Horn[19]	2.5	2	0.60	0.25	4
	2.5	0	0.13	0.33	5

2.2.2 DIRECT SOLAR ENERGY ABSORPTION BY THE BALLOON WALL

Q2 = CQ2 * AV * FV * HC ,

where

CQ2 is a correction coefficient to be established on the basis of experience (initially, CQ2 = 1),

AV is the effective UV absorptance of the film,

FV is the effective UV flux.

This model of absorption of UV flux is essentially equivalent to that developed by Germeles and refined by Kreith and Kreider (see program comment cards for more detailed references). We believe that the use of a constant cross sectional shape is adequate for two reasons: (a) due to uncertainties in the actual shape, and (b) due to availability of the factor CQ2 for making necessary, small

adjustments. In any event, the shape of the cross section will be reasonably constant for any given short period of time.

2.2.3 IR ENERGY ABSORPTION BY THE BALLOON WALL

Q3 = CQ3 * AR * BZ * SR * TI**4 ,

where

- CQ3 is a correction coefficient to be established on the basis of experience (initially, CQ3 = 1),
- AR is the effective IR absorptance of the film,
- BZ is the Stefan-Boltzmann constant,
- SR is the effective IR receptor surface area, namely HC + (HC - SA) * LOG(0.00626/BE)/LGTEN,
- TI is the equilibrium radiation temperature,
- BE is the specific lift of the inflatant (a function of altitude),
- LOG( ) is the natural logarithm,
- LGTEN is the natural logarithm of 10 (converted to real in program).

This model is that of Horn and Carlson,[43] except for the definition of the effective surface area SR. This area varies from the entire surface area surrounding the gas bubble (at launch) to only the area of the horizontal cross-section of the gas bubble (at altitudes greater than 60,000 ft, approximately); this is consistent with Kreith and Kreider.[16] For altitudes up to 60,000 ft, the foregoing function defining SR is quite effective.

2.2.4 CONVECTIVE HEAT TRANSFER BETWEEN THE BALLOON WALL AND THE AIR, FREE AND FORCED

Q4 = CQ4 * CA * GN# * Nu * (Ta - Tf) * SA/DM ,

where

- CQ4 is a correction coefficient to be established on the basis of experience (initially, CQ4 = 1),
- CA is the air thermal conductivity coefficient, computed by subroutine VIRON,
- GN# is the Nusselt number correction, # = 2 (free), # = 3 (forced).

2.2.4.1 Free Convective Heat Transfer Between the Balloon Wall and the Air

For a balloon at rest (if such a case ever truly exists) the Nusselt number is, as in Section 2.2.1, of the general form:

Nu = a * [b + c * (Pr * Gr)**(d)] ,

where the Prandtl number for air is 0.72 and the Grashof number is expressed in this case as:

Gr = G * [(DM * SW/VS)**2] * ABS(Tf - Ta)/Ta ,

where

SW is the specific weight of air computed by subroutine VIRON,

VS is the viscosity of air computed by subroutine VIRON.

The values used for the constants a, b, c, and d, once again have not been consistent during the evolution of the performance models (see Table 2). As before, we use Carlson's constants in the proposed model; except that the value of "a" is included in the correction factor "GN2".

Table 2. Nusselt Number Model Constants for Free Heat Transfer Between the Balloon Film and the Air, Showing Similarities and Differences, and Arranged Chronologically

	a	b	c	d
Hall[9]	1	0	0.50	0.25
Germeles[13]	1	0	0.13	0.33
Fujii et al[15]	1	0	0.65	0.25
Kreith and Kreider[16]	1	2	0.60	0.25
Balis Crema et al[18]	1	0	0.56	0.25
Carlson and Horn[19]	1	2	0.60	0.25

2.2.4.2 Forced Convective Heat Transfer Between the Balloon Wall and the Air

For balloons in motion the Nusselt number is usually expressed in the general form:

Nu = b + c * (Re)**(d)

where b is an arbitrary constant (as are c and d) and Re is the Reynolds number. Here again, the values of the constants b, c, and d, have not been consistent during the evolution of the performance models (see Table 3). In the proposed model, we again use Carlson's constants, except that we make no exception for balloons with maximum volumes greater than 10 million ft^3. We believe that any such correction should await evaluation of the effects on the model output of other non-arbitrary changes.

Table 3. Nusselt Number Model Constants for Forced Heat Transfer Between the Balloon Film and the Air, Showing Similarities and Differences, and Arranged Chronologically. Notes reflect valid ranges for: (1) laminar flow, (2) turbulent flow, (3) 1.8 * 10E3 ≤ (Re) ≤ 1.4 * 10E5, (4) 0.4 * 10E5 ≤ (Re) ≤ 1.4 * 10E5, and (5) volumes greater than 19 * 10E6 ft^3

	b	c	d	Notes
Germeles[13]	0	0.37	0.60	
Fujii et al[15]	0	0.52	0.50	1
	0	0.03	0.80	2
Kreith and Kreider[16]	2	0.30	0.57	3
	2	0.41	0.55	4
Carlson and Horn[19]	0	0.37	0.60	
	0	0.74	0.60	5

2.2.5 IR ENERGY EMISSION BY THE BALLOON WALL

Q5 = CQ5 * ER * BZ * SA * Tf**4 ,

where

CQ5 is a correction coefficient to be established on the basis of experience (initially, CQ5 = 1),

ER is the effective IR emissivity of the balloon wall film.

Except for the fact that SA has been redefined in accordance with our new shape model, this is identical to the respective heat transfer model of Horn and Carlson.[43]

2.2.6 REFLECTED SOLAR ENERGY ABSORPTION BY THE BALLOON WALL

This equation is based on Eq. (47), Kreith and Kreider.[16]

Q6 = CQ6 * AV * (2 * HC) * GS * FF * RL * QA * f (AL, RE) ,

where

CQ6 is a correction coefficient to be established on the basis of experience (initially, CQ6 = 1),

AV is the effective UV absorptance of the film according to Carlson,

GS is the solar constant,

FF is the directional reflectivity factor according to Figure 15 of Kreith and Kreider,[16]

RL is the reflectance modeled after Figure 10, Kreith and Kreider,[16]

QA is the cosine of the solar zenith angle,

AL is the balloon altitude,

RE is the radius of the earth,

f( ) is the function: [1. - SQRT(AL/RE/2.)],

SQRT( ) is the FORTRAN notation for square root.

2.2.7 RADIATIVE EXCHANGE BETWEEN THE INFLATANT AND THE BALLOON WALL

Q7 = CQ7 * EI * BZ * SA * (Tg**4 - Tf**4) ,

where

CQ7 is a correction coefficient to be established on the basis of experience (initially, CQ7 = 1),

EI is the coefficient of radiative exchange between the inflatant and the balloon wall film.

The relationships expressed in the models of Q7 through Q11 are based on Carlson,[19] but in the default mode of our proposed model we reject Carlson's hypothesis; thus, terms Q7 through Q11 are set equal to 0.

2.2.8 DIRECT SOLAR ENERGY ABSORPTION BY THE INFLATANT

Q8 = CQ8 * AG * FV * HC ,

where

CQ8 is a correction coefficient to be established on the basis of experience (initially, CQ8 = 1),

AG is the effective coefficient of absorptivity of the inflatant in the UV.

2.2.9 REFLECTED SOLAR ENERGY ABSORPTION BY THE INFLATANT

Q9 = CQ9 * AG * (2 * HC) * GS * FF * RL * QA * f(AL, RE) ,

where

CQ9 is a correction coefficient to be established on the basis of experience (initially, CQ9 = 1).

2.2.10 IR ENERGY EMISSION BY THE INFLATANT

Q10 = CQ10 * EG * BZ * SA * Tg**4 ,

where

CQ10 is a correction coefficient to be established on the basis of experience (initially, CQ10 = 1),

EG is the effective IR emissivity of the inflatant.

2.2.11 IR ENERGY ABSORPTION BY THE INFLATANT

Q11 = CQ11 * EG * BZ * SR * TI**4 ,

where

CQ11 is a correction coefficient to be established on the basis of experience (initially, CQ11 = 1).

This relationship is taken directly from Carlson,[19] but it is necessary that we change his effective surface area term to be consistent with the assumption that the IR energy absorbed is dependent upon altitude (see Section 2.2.3).

3. MODELING THE AERODYNAMIC DRAG COEFFICIENT

3.1 General Considerations

It is improbable that one can develop an adequate model of balloon aerodynamic drag coefficients by statistical means alone. Any reasonably approximate, mathematical model must account for a number of hard-to-quantify phenomena and, therefore, might become quite complex. For example, the degree to which balloon envelopes are deformed by dynamic pressure due to vertical motion certainly depends on shell stresses relative to film yield stresses (which, in turn, depend on balloon film temperature). However, one might develop adequate approximations by analyzing separately the flight data for heavily loaded, moderately loaded, and lightly loaded balloon shells. On the other hand, one also might gain some important theoretical insights by studying relevant works on raindrops and parachutes (for example see Figure 14). In any case, one must always temper judgement with experience – consider the influence of gore pattern on balloon ascent configuration.*

We concluded earlier that the drag coefficient must depend on Reynolds number, Froude number, and, most probably, fractional volume – three dimensionless parameters. Given an unambiguous definition of the characteristic length, and accurate flight data (elapsed time, altitude, atmospheric temperature, and all initial flight conditions), we can compute the average value of each of these parameters for each increment of altitude; leaving only the corresponding drag coefficient to be determined.

*Considerations such as these might have made the statistical analyses of rise rates by Nolan and Keeney[45] more valuable to balloon users. Further, if they had used dimensionless terms, they could have reduced to a minimum the number of multiple regression analysis (MRA) terms – this would have improved the quality of their predictors.

45. Nolan, G. F., and Keeney, P. L. (1973) Analysis of Factors Influencing Rate of Rise of Large Scientific Balloons, AFCRL-TR-73-0753, AD 778070.

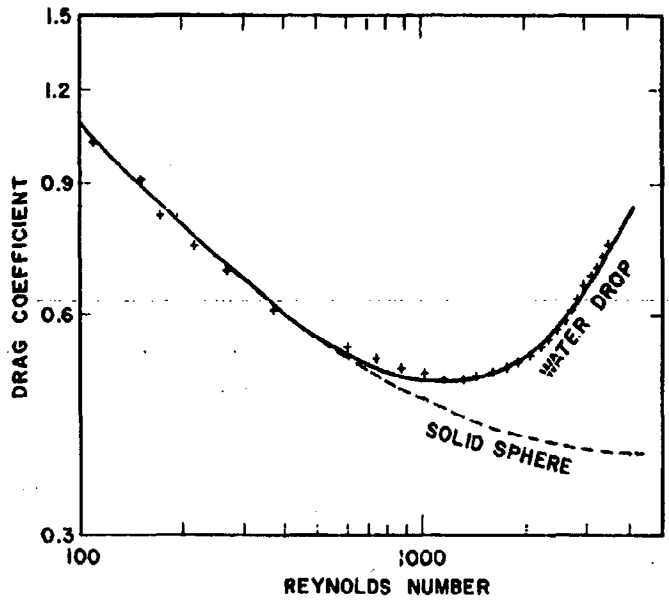

Figure 14. A Comparison of Reynolds Number vs Drag Coefficient for a Water Drop and a Sphere, According to Gillaspy.[37] By analogy the depicted relationship suggests that, for a given range of relative envelope stresses and for a given Froude number, the drag coefficient for an ascending balloon might be expected to increase with increasing Reynolds number – in some unspecified range. As in all of Section 3, we assume that the balloons are constructed from fully tailored gores (at least from gores that are fully tailored in the apex region)

Given the proposed comprehensive flight performance model (including the required sub-models, such as the atmosphere model, balloon shape model, and others), the problem of determining a drag coefficient is straightforward. We need only to establish an acceptable closure accuracy for the altitude computation, assume a drag coefficient, and then iteratively solve the model over each corresponding time interval until we find a drag coefficient value for which the altitude closure accuracy is satisfied. Other things being equal, we will then have a reasonably equally weighted set of four dimensionless variables for each altitude increment: a drag coefficient, Reynolds number, Froude number, and fractional volume. Usually, however, this will not be the case; unless, of course, we are unusually fortunate, or we have carefully selected flight data to account for qualitatively or imprecisely defined phenomena that significantly affect free balloon ascent rates.

3.2 Segregation of Data for Analysis

We have already presented in sufficient detail the effects of gore patterns on balloon ascent configurations. We are forced to conclude therefrom that fully tailored balloons and cylinder type balloons will have vastly different drag coefficient models. Thus, because cylinder balloon types (full cylinder and semi-cylinder) are infrequently used and even more infrequently manufactured for new programs, only fully tailored balloons should be considered in our present effort to model free-balloon drag coefficients.

Thermodynamic phenomena are covered by sub-models included in the proposed flight performance model. However, there is some concern about the accuracies of the assumptions on which present (and prior) heat transfer processes are based. Because of this, we seek to make our drag coefficient solutions as independent as is possible of thermodynamic considerations. The first and obvious choice is to use only data from flights launched at night. In this way, thermodynamic inaccuracies arise only from assumptions about the infrared, and conductive heat transfer models. Errors in the models dealing with direct and reflected solar radiation are eliminated.

A natural transition point both in the vertical motion of the balloon and in the dynamic, mechanical responses of the balloon envelope material, occurs near the tropopause. Both of these results are due to the reversal of the ambient temperature gradient. The former phenomenon, the slowing of the ascent rate, is well understood, and quite thoroughly documented. The latter phenomenon has generally been associated with balloon bursts due to cold brittleness of the polyethylene film; its implications with respect to both subsequent balloon failures, and altered resistance to ascent shape deformation are not well understood. With respect to shape deformability, we believe that the noted relaxation of the strain in the envelope material in the crown of the balloon[46] supports the contention that, for a balloon ascending above the tropopause, relative stress (actual stress divided by yield stress) changes. Thus, the shape can deform more easily. Therefore, we suggest that flight data above and below the tropopause be segregated for the purpose of drag coefficient modeling. The least advantage of this approach will be the existence of a logically distinct set of data by which a developed model may be tested. (When the tropopause temperature is different than the minimum temperature, prudent judgement is required.)

Finally, we recommend that flights be separated into two other classes: heavily loaded and light or moderately loaded balloons. To accomplish this, stress indices such as those used by the NSBF or AFGL (see Dwyer[42]) should be adequate.

---

46. Rand, J. L. (1982) Balloon Film Strain Measurements, Workshop on Instrumentation and Technology for Scientific Ballooning, XXIV COSPAR Plenary Meeting, Ottawa, Canada, 16 May - 2 June 1982.

3.3 Modeling Using Multiple Regression Analysis

Those familiar with modeling with MRA concede that it is more an art than a science, but, having decided to use dimensionless variables, we have taken the first and biggest step in our analysis in a sound scientific manner. If we then segregate the input flight data as suggested, we will be taking the second step in a sound scientific manner; this will help to minimize variations due to other factors, as previously noted. Beyond this point (if one is to develop a practical model using MRA) it appears that one must rely on both mathematical art and scientific insight and, what is equally as important, care in the specification and collection of the input data.

4. REFINING THE THERMODYNAMIC COEFFICIENTS

It does not appear that one can easily, if at all, find conditions wherein the aerodynamic drag may be ignored. Consequently, the aerodynamic drag coefficient model must be developed before it is possible to introduce refinements of the models for direct and reflected solar radiation. As with development of the model for the aerodynamic drag coefficient, segregation of the input data is desirable if not imperative. However, the solution process does not appear to be so complicated; the general format of the models of the heat transfer processes are fairly well established and all that remains is (hopefully) to correct the coefficients.

5. CONCLUSIONS

We have laid a foundation for the development of a comprehensive flight performance model based on practical and theoretical considerations.

We have proposed that the aerodynamic drag coefficient model be based on three dimensionless variables: Reynolds number, Froude number, and fractional volume.

We have shown that:

(a) there has been a longstanding and widespread error in the definition of the instantaneous mass of the balloon film involved in the heat transfer processes,

(b) the gas bubble cannot be modeled realistically as a sphere,

(c) the gas bubble is asymmetrical except when it is at or very near its natural ceiling altitude,

(d) the actual gas bubble shape, and, most probably the added mass, is directly related to the type of gore pattern,

(e) theory does not support models of drag coefficient based on Reynolds number only,

(f) theory does support the use of the Froude number as one of the variables that affects the drag coefficient of a free balloon, and

(g) that fractional volume is a reasonable way to accommodate variations in overall balloon shape, consistent with the need to specify a characteristic length for use in establishing a reference drag area.

Finally, we have proposed a procedure for the analysis of actual flight data to enable the development of a practical, but also theoretically sound, model of the aerodynamic drag coefficient of a zero-pressure, free balloon, and subsequent refinement of the heat transfer models for direct and reflected solar energy.

References

1. Green, A. W. (1975) An approximation for the shapes of large raindrops, J. Appl. Meteorol. 14:1578-1583.
2. Clarke, E. T., and Korff, S. A. (1941) The radiosonde: The stratosphere laboratory, J. Franklin Inst. 232(No. 4).
3. University of Minnesota (1952) Progress Report on Research and Development in the Field of High Altitude Balloons, Volume V, Contract Nonr-710(01), p. VI-181.
4. Ney, E. P., and Winckler, J. R. (1955) Prediction of Ascent Rates for Plastic Balloons, Technical Report, Contract No. AF19(604)-1135, Tufts University, Medford, Mass.
5. Erickson, M., and Froehlich, H. (1958) Vertical Motion of a Free Balloon, General Mills, Inc., Report No. 1858.
6. Nelson, J. R. (1974) Criteria for Controlling Vertical Motion of Stratospheric Balloons, Final Report, Contract N00014-73-C-0382, Winzen Research, Inc.
7. Kremser, G. (1976) On the flight behavior of stratospheric balloons, Space Sci. Inst. 2:489-498.
8. Smith, J. R., and Murray, W. D. (1949) Constant Level Balloons, Technical Report No. 93.02, Contract W28-099-ac-241, College of Engineering, New York University, pp. 34-40, 58, 62.
9. Hall, N. A. (1952) Elements of Balloon Dynamics, Topical Report on Project Gopher, Contract AF33(600)-6298, General Mills, Inc., pp. 1, 2.
10. University of Minnesota (1952) Progress Report on Research and Development in the Field of High Altitude Balloons, Volume I, Contract Nonr-710(01), pp. 4-23, 4-37.
11. Emslie, A. G. (1961) Balloon Dynamics, Technical Report I, Contract No. NONR-3164(00), Arthur D. Little Inc., pp. 13, 29, 30.

12. Dingwell, I. W., et al (1963) Vertical Motion of High Altitude Balloons, Technical Report II, Contract No. NONR-3164(00), Arthur D. Little, Inc., p. 5.

13. Germeles, A. E. (1966) Vertical Motion of High Altitude Balloons, Technical Report IV, Contract No. NONR-3164(00), Arthur D. Little, Inc.

14. Hansen, R. (1970) Vertical Simulation of High Altitude Balloons, Technical Report, Contract No. F19628-69-C-0213, G. T. Schjeldahl Co.

15. Fujii, M., et al (1973) A thermal analysis of the vertical motion of balloons, Proc. Tenth International Symposium on Space Technology and Science, pp. 1235-1243.

16. Kreith, F., and Kreider, J. F. (1971) Numerical Prediction of the Performance of High Altitude Balloons, Report No. NCAR-TN/STR-65, National Center for Atmospheric Research (Revised, 1974).

17. Romero, M., et al (1978) Prevision et depouillements de vols de ballons ouverts en descente pilotee, Proceedings of the European Sounding-Rocket, Ballon and Related Research With Emphasis on Experiments at High Latitudes.

18. Balis Crema, L., et al (1978) Analisi Delle Traiettorie Dei Palloni Stratosferici Per Missioni Di Lunga Durata, Instituto di Tecnologia Aerospaziale, Universita di Roma (translation).

19. Carlson, L. A., and Horn, W. J. (1981) A Unified Thermal and Vertical Trajectory Model for the Prediction of High Altitude Balloon Performance, TAMRF Report No. 4217-81-01, Texas A & M University, NASA CR 156884.

20. Carlson, L. A., and Horn, W. J. (1981) A new thermal and trajectory model for high-altitude balloons, J. Aircraft 20(No. 6):500-507.

21. Emmons, G., et al (1950) Oscillations in the stratosphere and high troposphere, Bull. Am. Meteorol. Soc. 31(No. 4):135-138.

22. University of Minnesota (1952) Progress Report on Research and Development in the Field of High Altitude Balloons, Volume IX, Contract Nonr-710(01).

23. Hirsch, J. H., and Booker, D. R. (1966) Response of superpressure balloons to vertical air motions, J. Appl. Meteorol. 5:226-229.

24. Nishimura, J., and Hirosawa, H. (1968) The hunting mechanism of plastic balloons, ISAS Bull. 4(1B):93-110.

25. Morris, A. L., and Stefan, K. H. (1969) High Altitude Balloons as Scientific Platforms, National Center for Atmospheric Research.

26. Nishimura, J., et al (1971) Balloon behavior during level flight, ISAS Bull. 7(1C):257-268.

27. Levanon, N., et al (1974) On the behavior of superpressure balloons at 150 mB, J. Appl. Meteorol. 13:494-504.

28. Levanon, N., and Kushnir, Y. (1976) On the response of superpressure balloons to displacements from equilibrium density level, J. Appl. Meteorol. 15:346-349.

29. Julian, P., et al (1977) The TWERLE experiment, Bull. Am. Meteorol. Soc. 58(No. 9):936-948.

30. Massman, W. J. (1978) On the nature of vertical oscillations of constant volume balloons, J. Appl. Meteorol. 17:1351-1356.

31. Rivett, P. J. (1983) A world in which nothing ever happens twice, J. Oper. Rsch. Soc. 34(No. 8):681.

32. Brunt, D. (1927) The period of simple vertical oscillations in the atmosphere, Quart. J. Roy. Meteorol. Soc. 53:30-32.

33. Dwyer, J. F. (1978) Zero pressure balloon shapes, past, present, and future, Scientific Ballooning (COSPAR), W. Riedler, Ed., Pergamon Press, pp. 9-19.

34. Hoerner, S. F. (1965) Fluid-Dynamic Drag (published by author).

35. Dwyer, J. F. (1980) The Problem: Instantaneously Effecting Controlled Balloon-System Descent from High Altitude, AFGL-TR-80-0277, AD A100255.

36. Schlichting, H. (1968) Boundary-Layer Theory, McGraw-Hill Book Co., New York, 6th Edition, p. 16.

37. Gillaspy, P. H. (1981) Experimental Determination of the Effect of Physical Properties on the Drag of Liquid Drops, Ph.D. Dissertation, University of Nevada (funded under U. S. Army Research Office Contract No. DAAB29-77-G-1072).

38. Von Karman, T. (1945) Note on Analysis of the Opening Shock of Parachutes at Various Altitudes, A. A. F. Scientific Advisory Group.

39. Tryggvason, B. V., and Isyumov, N. (1978) Similarity requirements for inflatable structures, Proceedings of the Third U. S. National Conference (on Wind Engineering Research), University of Florida, Gainesville, Florida, pp. 335-338.

40. Tryggvason, B. V. (1979) Aeroelastic modelling of pneumatic and tensioned fabric structures, Proceedings of the Fifth International Conference (on Wind Engineering), Fort Collins, Colorado, pp. 1061-1072.

41. Landau, L. D., and Lifshitz, E. M. (1959) Fluid Mechanics, Addison-Wesley Publishing Co., Inc., p. 63.

42. Dwyer, J. F. (1982) Polyethylene Free Balloon Design From the Perspective of User and Designer, AFGL-TR-82-0350, AD A127553.

43. Horn, W. J., and Carlson, L. A. (1983) THERMTRAJ: A Fortran Program to Compute the Trajectory and Gas Film Temperatures of Zero Pressure Balloons, NASA Contractor Report 168342.

44. Carlson, L. A. (1979) A new thermal analysis model for high altitude balloons, Proceedings, Tenth AFGL Scientific Balloon Symposium, Catherine L. Rice, Ed., pp. 187-206.

45. Nolan, G. F., and Keeney, P. L. (1973) Analysis of Factors Influencing Rate of Rise of Large Scientific Balloons, AFCRL-TR-73-0753, AD 778070.

46. Rand, J. L. (1982) Balloon Film Strain Measurements, Workshop on Instrumentation and Technology for Scientific Ballooning, XXIV COSPAR Plenary Meeting, Ottawa, Canada, 16 May - 2 June 1982.

Bibliography

Carlson, L. A. (1979) THERMNEW - A Preliminary Model and Computer Program for Predicting Balloon Temperatures at Float, Contract NCAR 1-73, Texas A&M University.

Carlson, L. A., and Horn, W. J. (1981) A New Thermal and Trajectory Model for High Altitude Balloons, AIAA 7th Aerodynamic Decelerator and Balloon Technology Conference, San Diego, Calif.

Davidson, A. R. Jr. (1966) Vertical Flight Profile Analysis of Stratoscope II Flight S-5, Vitro Corporation, Technical Note 1745.50-1.

Dingwell, I. W. (1967) Thermal Radiation Properties of Some Polymer Balloon Fabrics, Technical Report VI, Contract No. Nonr-3164(00), Arthur D. Little, Inc.

Fujimoto, Y. (1966) The temperature of the gas in a balloon in relation to the ascending velocity, ISAS Bull. 2(1C):407-408.

Greenfield, S. M., and Davis, M. H. (1963) The Physics of Balloons and Their Feasibility as Exploration Vehicles on Mars, The Rand Corporation, R-421-JPL.

Gunn, R., and Kinzer, G. D. (1949) The terminal velocity of fall for water droplets in stagnant air, J. Meteorol. 6:243-248.

Hoecker, W. H., and Hanna, S. R. (1971) Computed Response of Tetrahedral Constant-Density Balloons to Vertical Sinusoidal and Helical Air Motions, NOAA Technical Memorandum, ERL ARL-31.

Kawamura, R. (1966) A study of the aerodynamics of balloons, ISAS Bull. 2(1C): 398-402.

Kreider, J. F., and Kreith, F. (1975) Numerical prediction of high altitude zero-pressure balloon vertical motion, J. Heat Transfer 97:155-157.

Kreith, F. (1970) Thermal design of high-altitude balloons and instrument packages, J. Heat Transfer 92:307-332.

Kreith, F. (1975) Energy balance and a flight model, Scientific Ballooning Handbook, Section III, National Center for Atmospheric Research.

LaPadula, C. D., and Polcaro, C. F. (1977) A Mathematical Model of [A] Transcontinental Balloon, XXVIII Congress, International Astronautical Federation, IAF-77-167.

Lucas, R. M., and Hall, G. H. (1966) The Measurement of High Altitude Balloon Gas Temperature, Technical Report V, Contract No. Nonr-3164(00), Arthur D. Little, Inc.

Lucas, R. M., et al (1970) Experimental balloon gas and film temperatures, Proceedings, Sixth AFCRL Scientific Balloon Symposium, L. A. Grass, Ed., pp. 227-240.

McDonald, J. E. (1954) The shape and aerodynamics of large raindrops, J. Meteorol. 11:478-494.

Nelson, J. R. (1975) Criteria for Controlling Vertical Motion of Stratospheric Balloons, Final Report on Contract N00014-75-C-0072, Winzen Research, Inc.

Nelson, J. R. (1975) Two Balloons Fly Together, Flight Report, Issue No. 1, Winzen Research, Inc.

Ney, E. P. (1951) Preliminary Notes on Factors Influencing Long Duration Flights, University of Minnesota.

Nishimura, J., and Hirosawa, H. (1969) On the hunting mechanism of the plastic balloon, Proceedings of the Eighth International Symposium on Space Technology and Science, pp. 1157-1161.

Nishimura, J., et al (1973) Effect of heat transfer on the motion of a balloon at high altitudes, ISAS Bull. 9(1B):167-185.

Oakland, L. (1963) Summary Report on Balloon Physics as Related to Long Duration Flights, Flight Report, Issue No. 1, Winzen Research, Inc.

Pommereau, J. P., and Hauchecorne, A. (1974) Project Transatlantique Premiere Partie: Essais En Vol Et Interpretation, Service D'Aeronomie, C.N.R.S.

Reynolds, R. D. (1964) Investigation of the Effect of Lapse Rate on Balloon Ascent Rate, U. S. Army Report No. ERDA-140.

Shames, I. H. (1962) Mechanics of Fluids, McGraw-Hill Book Co., New York, pp. 35, 108, 484.

Smith, D. R. L., and Houghton, J. (1960) Fluid Mechanics Through Worked Examples, Cleaver-Hume Press Ltd., London, pp. 276, 287.

Spilhaus, A. F. (1948) Raindrop size, shape and falling speed, J. Meteorol. 5:108-110.

Appendix A

Software for Developing, Verifying, and Using Aerodynamic and Thermodynamic Constants and Models

A1. PROGRAM FINDCD

Program FINDCD is a FORTRAN coded set of routines and models discussed in the main text; it is written to run on an IBM PC and has a compiled executable version, QCD.EXE. This program, working on the assumptions that all of the thermodynamic models are sufficiently accurate, collects for each point in any chosen flight profile the values of drag coefficient CD, Reynolds Number RN, Froude Number FRN, and fractional volume VB.

A1.1 Program Logic

The logic by which the values of the terms CD, RN, FRN, and VB are determined is shown in Figures A1 and A2. Fundamentally, it is an iterative method of adjusting the value of CD between each successive set of points until the actual and computed altitudes are satisfactorily close for the related elapsed time.

A1.2 Strategy

It appears that the accuracy of related aerodynamic terms can be enhanced by initially restricting analyses to flights launched and ascending in darkness — this eliminates potential errors due to solar energy input models. Eventually one will have to alter this program to accommodate deballasting sequences — not at all a difficult task. This will be required because on most high-altitude night flights deballasting is required to maintain ascent rates compatible with mission profiles.

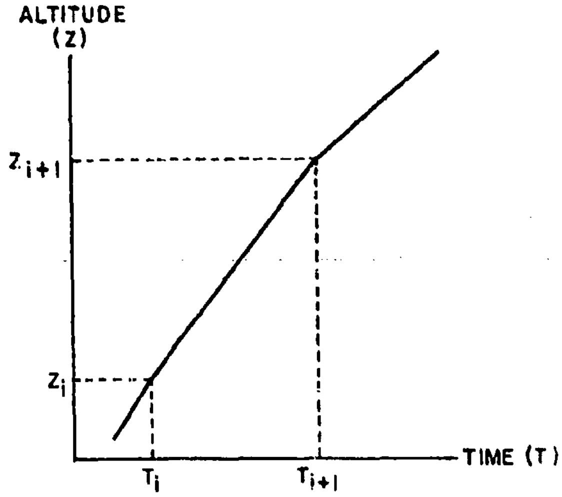

Figure A1. Representative Segment of Vertical Flight Profile

A1.3 An Aerodynamic Drag Coefficient Model

The definition of an adequate drag coefficient model is still to be found. If physical modeling is an art then mathematical modeling, being one step beyond, might be considered a black art. There is some discussion – both in the main text and in the notes imbedded in the program comments – regarding the use of MRA as a method; doubtlessly the data developed by this program will suggest more explicit approaches.

A2. PROGRAM FROUDE

Program FROUDE is also a set of FORTRAN coded routines and models discussed in the main text; it too is written to run on an IBM PC and has a compiled executable version, FLITE.EXE. It assumes the existence of a drag coefficient model in the following format:

```
      i=20
CD = SUM [Ai * (FR**C1i) * (RN**C2i) * (VB**C3i)] .
      i=1
```

This model format was selected by the author as one easily adapted to MRA modeling, one suitable for expressing simple series models, and one with the inherent capacity to express quite complex relationships. Like FINDCD, this program permits interactive alteration of most of the various model coefficients. Consequently,

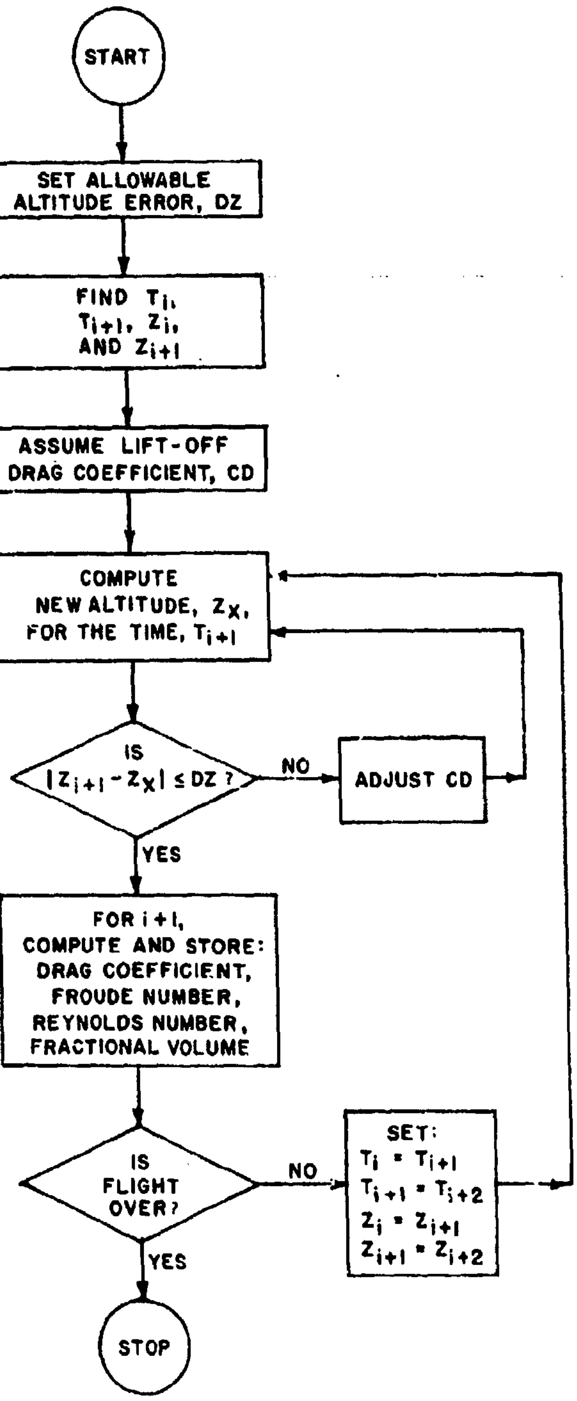

Figure A2. Elementary Logic of Program FINDCD

it can be used for either flight performance prediction or for analysis of performance sensitivity to changes in particular coefficients.

A3. INPUT FORMATTING

There are two additional FORTRAN programs, CD.EXE and QCDDATA.EXE, written to format and store the required input files for FLITE.EXE and QCD.EXE.

A3.1 CD.EXE

This program supports only FLITE.EXE; it stores the CD model coefficients and exponents in the required format.

A3.2 QCDDATA.EXE

This input file formatting program supports both FLITE.EXE and QCD.EXE. It has one particularly interesting feature; it distinguishes between radar flight data and altitude translated from a standard altitude table. In the latter case it provides the altitude corrected for the local atmospheric temperature profile and launch site pressure.

```
      PROGRAM FINDCD
*************************************************************************
*                                                                       *
*   PROGRAM: FINDCD                                     20 FEB 1985     *
*                                                                       *
*   THE EXECUTABLE VERSION OF THIS PROGRAM IS DESIGNATED AS 'QCD'       *
*   WHICH HAS BEEN COMPILED UNDER MICROSOFT FORTRAN77 TO BE RUN ON      *
*   AN IBM PC.                                                          *
*                                                                       *
*   THIS PROGRAM IS USED TO DETERMINE THE REYNOLDS NO., FROUDE NO.,     *
*   FRACTIONAL VOLUME AND RELATED AERODYNAMIC DRAG COEFFICIENT FOR      *
*   WHICH THE MODEL-PREDICTED ASCENT RATE AND ACTUAL FLIGHT ASCENT      *
*   RATE AGREE WITHIN A GIVEN TOLERANCE, OVER A REASONABLY LARGE        *
*   ALTITUDE SPAN. THE RESULTS ARE INTENDED TO BE USED AS INPUT IN      *
*   A MULTIPLE REGRESSION ANALYSIS TO PROVIDE A MODEL OF THE DRAG       *
*   COEFFICIENT AS A FUNCTION OF REYNOLDS NO., FROUDE NO., AND THE      *
*   FRACTIONAL VOLUME. THE DESIRED OUTPUT VARIABLE VALUES ARE FOUND     *
*   BY ITERATIVELY ADJUSTING THE ASSUMED DRAG COEFFICIENT UNTIL         *
*   THE GIVEN TOLERANCE IS ACHIEVED. ENHANCED ACCURACY IS ACHIEVED      *
*   BY REDUCING THE VARIABILITY OF FACTORS INFLUENCING THE ASCENT       *
*   RATE; PRIMARILY, THIS IS EFFECTED BY USING NIGHTTIME ASCENT DATA    *
*   UNAFFECTED BY SOLAR INPUT.                                          *
*                                                                       *
*   THE USE OF MRA TO ACHIEVE THE MODEL IS ONLY ONE APPROACH, AND       *
*   ONE SHOULD NOTE: 1) THAT THE USE OF MRA IS VERY MUCH AN ART, AND    *
*   2) THAT THE RESULTING MATHEMATICAL MODEL MAY BE SIGNIFICANTLY       *
*   DIFFERENT THAN THE TRUE PHYSICAL MODEL. CAUTION IS URGED IF THE     *
*   RESULTING FORM OF THE MRA MODEL IS TO BE USED TO PREDICT OTHER      *
*   PHYSICAL RELATIONSHIPS.                                             *
*                                                                       *
*   FOR COMMENTS ON SPECIFIC LINES OF CODE, SEE PROGRAM FROUDE.         *
*                                                                       *
*   THIS PROGRAM WAS DEVELOPED AT THE AIR FORCE GEOPHYSICS LABORATORY   *
*   AS PART OF IN-HOUSE WORK UNIT NO. 76591114                          *
*                                                                       *
*************************************************************************

      COMMON CA,DTI,E(30,2),PE,PO,RO,SW,TI,TIR1,TIRO,TR,TROP,VS

      DIMENSION A(5),B(5),C(5),D(8,5),Y(8,6),Q(8,6),FLY(100,2),FR(100,4)

      CHARACTER*82 FINN,FOUT,FNAME,HEAD1,HEAD2

      DATA A/3.141592654,.5,.2928932,1.7071068,.1666666666/,GS/96./
      DATA B/O.,2.,1.,1.,2./,C/.01745329252,.5,.2928932,1.7071068,.5/
      DATA BZ/3.6995E-10/,G/32.1741/,RE/20855278./,RA/53.35/,RG/386.076/
      DATA DTM/20./,DTV/0.5/,DT/8./,JF,LL2,LL3/3*1/,LAUNCH,LEAP/2*0/
      DATA AC3,AC4,DB,DDO,DD1,T,TT,TTT,VD,VT,VV,Q7,Q8,Q9,Q10,Q11/16*0./
      DATA ALF/1.83E-07/,BET/.682/,GAM/1443./,CV/586.73/,CF/428./
      DATA AYV/.001/,RYV/.114/,TYV/.885/,AYR/.031/,EYR/.031/,RYR/.127/
      DATA TYR/.842/,AYRG/.0020/,WOW/.0048/,CM/.5/,VL/.01/
      DATA CQ1,CQ2,CQ3,CQ4,CQ5,CQ6,CQ7,CQ8,CQ9,CQ10,CQ11/11*1./
      DATA GN1,GN2,GN3/3*1./
```

```
      HEAD1='              RN              FRD             VB             CD'
      HEAD2='            TIME             ALT          SPEED          ERROR'
 1    FORMAT(A)
 2    FORMAT(1X,'ENTER NAME OF INPUT FILE; B:filespec.FLY      ',\)
 3    FORMAT(1X,'AND NAME OF OUTPUT FILE; B:filespec.JFD      ',\)
 4    FORMAT(13(E15.8/),E15.8)
 5    FORMAT(2F9.3)
 6    FORMAT(2F8.0)
 7    FORMAT(1X,4E13.7)
 8    FORMAT(4(E15.8/),E15.8)
 9    FORMAT(I3)
 10   FORMAT(10(E15.8/),E15.8)
 11   FORMAT(A,I3,A)
 12   FORMAT(6E15.8)
 13   FORMAT(1X,2E13.7)
 14   FORMAT(2E13.7)
 15   FORMAT(3E13.7)
 17   FORMAT(1X,3E13.7)
 18   FORMAT(2X,A)

      CALL INFORM

      OPEN(5,FILE='LPT1',FORM='FORMATTED')

 50   WRITE(*,1)' DEFAULT GAS & FILM VALUES ?  [ 0/1 = N/Y ]'
      READ(*,9)I
      IF (I.EQ.0) THEN
        I=1
        WRITE(*,1)' INSERT APPROPRIATE DATA DISK IN B-DRIVE AND'
        WRITE(*,1)' ENTER GAS & FILM FILE NAME, B:filespec.GAF'
        READ(*,1)FNAME
        OPEN(3,FILE=FNAME,FORM='FORMATTED')
        READ(3,4)ALF,BET,GAM,CV,CF,WOW,AYV,RYV,TYV,AYR,EYR,RYR,TYR,AYRG
        CLOSE(3)
        WRITE(5,1)FNAME
        WRITE(*,1)' '
      ENDIF
      IF (I.NE.1) GOTO 50

 60   WRITE(*,1)' DEFAULT HEAT XFER COEFs., VIRTUAL MASS COEF. AND'
      WRITE(*,1)' EFFECTIVE ZERO ASCENT RATE ?  [ 0/1 = N/Y ]'
      READ(*,9)I
      IF (I.EQ.0) THEN
        I=1
        WRITE(*,1)' INSERT APPROPRIATE DATA DISK IN B-DRIVE AND'
        WRITE(*,1)' ENTER COEFFICIENT FILE NAME, B:filespec.CMV'
        READ(*,1)FNAME
        OPEN(3,FILE=FNAME,FORM='FORMATTED')
        READ(3,10)CQ1,CQ2,CQ3,CQ4,CQ5,CQ6,CQ7,CQ8,CQ9,CQ10,CQ11
        READ(3,8)GN1,GN2,GN3,CM,VL
        CLOSE(3)
        WRITE(5,1)FNAME
        WRITE(*,1)' '
      ENDIF
      IF (I.NE.1) GOTO 60
```

```
      WRITE(*,1)' INSERT THE FLIGHT DATA DISK IN B-DRIVE AND'
      WRITE(*,2)
      READ(*,1)FINN
      WRITE(*,1)' '
      WRITE(*,3)
      READ(*,1)FOUT
      WRITE(*,1)' '
      OPEN(3,FILE=FINN,FORM='FORMATTED')
      READ(3,1)FNAME
      READ(3,4)SIG,GL,CLEN,AD,VTM,THK,TCP,WB,WF,WT,FL,DBB,SPD,PO
      READ(3,4)CD,ERR,DU3,DCD,DU5,AGAS,ELL,GH,XD,XG,XL,TS,XIN,XINN
      DO 100 I=1,30
100   READ(3,5)E(I,1),E(I,2)

C     OPTION: SELECTION OF ANALYSIS CUT-OFF POINT
110   INN=INT(XINN)
      WRITE(*,11)' THERE ARE',INN,' FLIGHT DATA POINTS.'
      WRITE(*,1)' ENTER NUMBER TO BE ANALYZED.'
      READ(*,9)INN
      IF (INN.GT.INT(XINN)) GOTO 110
      DO 120 I=1,INN
120   READ(3,6)FLY(I,1),FLY(I,2)
      CLOSE(3)

      WRITE(*,1)' ENTER DIFFERENTIAL GROWTH LIMITS: ALT. [ ft ],'
      WRITE(*,1)' ASCENT RATE [ ft/sec ] AND TEMP. [ deg R ]'
      READ(*,15)ALIM,VLIM,TLIM
      WRITE(*,1)' '

      WRITE(*,1)' TO CHANGE DT [max] OR DT [vent] ENTER NEW NON-ZERO'
      WRITE(*,1)' VALUE, OTHERWISE ZERO. ENTER ZERO OR ONE FOR AGAS.'
      WRITE(*,1)' DTM, DTV, AGAS [ gas absorbs in IR, 0/1 = N/Y ]'
      WRITE(*,14)DTM,DTV,AGAS
      READ(*,15)GG1,GG2,AGAS
      IF (GG1.GT.0.) DTM=GG1
      IF (GG2.GT.0.) DTV=GG2
      WRITE(*,1)' '

C     OPTION: ORIGINAL OR REVISED DRAG COEFFICIENT ESTIMATE AND DRAG
C     COEFFICIENT INCREMENT FOR ITERATIVE ADJUSTMENT
      WRITE(*,1)' TO CHANGE CD OR DCD ENTER NEW VALUE, OTHERWISE ZERO.'
      WRITE(*,13)CD,DCD
      READ(*,14)GG1,GG2
      IF (GG1.GT.0.) CD=GG1
      IF (GG2.GT.0.) DCD=GG2
      WRITE(*,1)' '

      WRITE(5,18)FOUT
      WRITE(5,1)' '
      WRITE(5,1)' DTM  DTV  AGAS  ALIM  VLIM  TLIM'
      WRITE(5,7)DTM,DTV,AGAS,ALIM,VLIM,TLIM
      WRITE(5,1)' '
      WRITE(5,1)' CD  DCD'
      WRITE(5,13)CD,DCD
      WRITE(5,1)' '
```

```
      LGTEN=LOG(10.)
      IN=INT(XIN)
      LL1=IN
      LLO=IN


      EGAS=AGAS
      AYRG=AGAS*AYRG
      AYRG1=1.-AYRG
      AV=AYV*(1.+TYV*AYRG1/(1.-RYV*AYRG1))
      AG=AYRG*TYV/(1.-RYV*AYRG1)
      DBB=DBB/60.
      WTX=0.
      SPD=1.69*SPD
      DD4=PO
      DD5=PO

      DO 190 I=2,30
      TROP = E(I-1,1)
      IF (E(I,2).GE.E(I-1,2)) GOTO 200
190   CONTINUE
200   TIRO=1.8*E(1,2)-5.55
      TIR1=.74*TIRO
      DTI=-.26*TIRO/(TROP-E(1,1))

      IF (XL.GE.20.) GOTO 300
      RL=-.0025*XL+.15
      GOTO 600
300   IF (XL.GE.30.) GOTO 400
      RL=.1
      GOTO 600
400   IF (XL.GE.40.) GOTO 500
      RL=.005*XL-.05
      GOTO 600
500   RL=.0075*XL-.15

600   XD=C(1)*XD
      XL=C(1)*XL

      CX=COS(XL)*COS(XD)
      SX=SIN(XL)*SIN(XD)

      CALL UPSON(1,SIG,ABL)
      GB=CLEN/GL
      CALL NELSON(U0,GB,SIG)
      WC=U0*ABL*WOW*TCP*GL*GL

      DO 700 J=1,5
      DO 700 I=1,8
      Y(I,J)=0.
700   Q(I,J)=0.
```

```
      Y(1,1)=0.
      Y(2,1)=WP+WT
      Y(3,1)=ELL
      CALL VIRON(ELL,BP,DD1,DD3,DD5,LL1,LL3,DD7,DD9,TK,1)
      BE=(1.-.138185)*SW
      V0=(1.+FL)*(WB+WP+WT)/BE
      Y(4,1)=TR
      Y(5,1)=V0*PE/TR/RG
      Y(6,1)=TR
      Y(7,1)=V0
      Y(8,1)=PE

++++++++++++++++++++++++ BEGIN RUNGE-KUTTA ITERATION +++++++++++++++++++

800   DO 1600 KJ=2,5
      KK=KJ-1
      CALL VIRON(Y(3,KK),BP,DD1,DD3,DD5,LL1,LL3,DD7,DD9,TK,1)
      BE=(1.-.138185*TR/Y(6,KK))*SW

       IF (WT.NE.WTX) THEN
         X=LOG(1.+WB/(WT+WP))
         CALL UPSON(2,X,SIGX)
         CALL UPSON(3,SIGX,VU)
         VT=VU*GL**3
         WTX=WT
          IF (VT.GT.VTM) VT=VTM
       ENDIF

        IF (KK.EQ.1) THEN
          VB=Y(7,KK)/VT
          CALL MYBLN(VB,GL,GB,GN,DH,SB,DM,RM,HC)
          U=U0
           IF (GN.LT.CLEN) CALL NELSON(U,GB,SIG)
          WE=SB*(WB-WC)+WDW*U*ABL*TCP*GL*GL
          SA=SB*ABL*GL*GL
          SR=HC
           IF (BE.GE.0.00626) SR=HC+(HC-SA)*LOG(.00626/BE)/LGTEN
        ENDIF

        PG=DH*BE
        SWG=Y(5,KK)/Y(7,KK)
        VSG=ALF*Y(6,KK)**BET
        CG=GAM*VSG
        SPEED=Y(1,KK)

        IF (LAUNCH.EQ.0) SPEED=SPD

        SPDSQ=SPEED*SPEED
        RN=DM*SW*ABS(SPEED)/VS
        BUOY=SW*Y(7,KK)
        WS=Y(2,KK)+Y(5,KK)+WB
        FORCE=BUOY-WS
        DRAG=.5*RO*CD*HC*SPEED*ABS(SPEED)
```

```
      D(1,KJ)=G*(FORCE-DRAG)/(WS+CM*BUOY)
      IF (LAUNCH.EQ.0) D(1,KJ)=0.
      D(2,KJ)=-DB
      D(3,KJ)=Y(1,KK)

C++++++++++++++++++++++++ BEGIN HEAT TRANSFER RATES ++++++++++++++++++++

      EYRG=EGAS*3.42E-06*(Y(6,KK)/1.8)**.815?

      EYRG1=1.-EYRG
      ER=EYR*(1.+TYR*EYRG1/(1.-RYR*EYRG1))
      AR=ER
      EI=EYRG*EYR/(1.-RYR*EYRG1)
      EG=EYRG*TYR/(1.+RYR*EYRG1)

      CALL PRGR(GP,.67,DM,SWG,Y(4,KK),Y(6,KK),VSG)
      IF (GP.LT.15E+07) GNU=2.+.6*GP**0.25
      IF (GP.GT.15E+07) GNU=.13*GP**(1./3.)

      Q1=CQ1*GN1*SA*(Y(6,KK)-Y(4,KK))*CG*GNU/DM

      QA=SX+CX*COS(C(1)*(GH-XG+T/240.))

      ARMS=(BP/PO)*(SQRT(1228.6+376750.44*QA*QA)-613.8*ABS(QA))

      TRM1=.5*(EXP(-.65*ARMS)+EXP(-.095*ARMS))

      FV=GS*CS*TRM1
      IF (QA.LT.0.) THEN
        FV=0.
        QB=-SQRT(1.-(RE/(RE+Y(3,KK)))**2)
        IF (QB.LT.QA) THEN
          ZZ=SQRT(1.-QA*QA)*(RE+Y(3,KK))-RE
          CALL VIRON(ZZ,PAM,DD0,DD2,DD4,LL0,LL2,DD6,DD8,TX,2)
          AM=35.1*PAM/PO
          TRM2=.5*(EXP(-.65*AM)+EXP(-.095*AM))
          FV=GS*CS*TRM2*TRM2/TRM1
        ENDIF
      ENDIF

      Q2=CQ2*AV*FV*HC
      Q3=CQ3*AR*BZ*SR*TI**4

      IF (ABS(SPEED).LT.VL) THEN
        CALL PRGR(GP,.67,DM,SW,Y(4,KK),TR,VS)
        GNU=GN2*(2.+.6*GP**0.25)
      ELSE
        GNU=.37*GN3*RN**0.6
      ENDIF

      Q4=CQ4*SA*(TR-Y(4,KK))*CA*GNU/DM
      Q5=CQ5*ER*BZ*SA*Y(4,KK)**4
```

```
      ZS=57.29578*ATAN(SQRT((1.-QA*QA)/(QA*QA)))
      FF=1.
      IF (ZS.GT.77.2) FF=.09375*ZS-5.4375
      IF ((FF.EQ.1.).AND.(ZS.GT.25.)) FF=.0153*ZS+.6169

      Q6=CQ6*AV*(2.*HC)*GS*FF*RL*(1.-SQRT(Y(3,KK)/RE/2.))*QA

      IF (EI.NE.0.) THEN
        Q7=CQ7*EI*BZ*(Y(6,KK)**4-Y(4,KK)**4)*SA
        Q8=CQ8*AG*FV*HC
        Q9=CQ9*AG*(2.*HC)*GS*FF*RL*(1.-SQRT(Y(3,KK)/RE/2.))*QA
        Q10=CQ10*EG*BZ*SA*Y(6,KK)**4
        Q11=CQ11*EG*BZ*SR*TI**4
      ENDIF

++++++++++++++++++++++++ END HEAT TRANSFER RATES +++++++++++++++++++++

      D(4,KJ)=(Q1+Q2+Q3+Q4-Q5+Q6+Q7)/CF/WE

      IF ((VV.GT.0.).AND.(KK.EQ.1)) THEN
        IF (PG.GE.0.936) VV=.559-PG/8
        IF (PG.LT.0.936) VV=.72222*SQRT((1.872-PG)*PG)
        VV=6.127*VV*SQRT(SW/SWG-1.)
      ENDIF

      IF(VD.LE.0.) IVENT=0

      IF (IVENT.EQ.0) THEN
        D(5,KJ)=-VV*SWG
        D(6,KJ)=(Q8+Q9-Q1-Q7-Q10+Q11-SW*Y(7,KK)*D(3,KJ))/Y(5,KK)/(CV+RG)
        D(7,KJ)=Y(7,KK)*(D(5,KJ)/Y(5,KK)+D(6,KJ)/Y(6,KK)+D(3,KJ)/RA/TR)
        D(8,KJ)=-SW*D(3,KJ)
      ELSE
        IF (LEAF.EQ.0) DT=DTV
        VD=AD*SQRT(2.*G*ABS(Y(8,KK)-PE)/SWG)
        D(5,KJ)=-(VV+VD)*SWG
        D(6,KJ)=(Q8+Q9-Q1-Q7-Q10+Q11-PE*D(5,KJ)/SWG)/Y(5,KK)/CV
        D(7,KJ)=0.
        D(8,KJ)=Y(8,KK)*(D(5,KJ)/Y(5,KK)+D(6,KJ)/Y(6,KK))
      ENDIF

      DO 1500 KI=1,8
      QQ=A(KJ)*(D(KI,KJ)-B(KJ)*Q(KI,KK))
      Y(KI,KJ)=Y(KI,KK)+DT*QQ
 1500 Q(KI,KJ)=Q(KI,KK)+3.*QQ-C(KJ)*D(KI,KJ)
 1600 CONTINUE

++++++++++++++++++++++ END RUNGE-KUTTA ITERATION +++++++++++++++++++++

      IF (Y(7,5).GT.VT) THEN
        VT=Y(7,1)
        IVENT=1
        GOTO 800
      ENDIF
```

```
      IF (IVENT.EQ.0) THEN
        IF (ABS(Y(1,5)-Y(1,1)).GT.VLIM) GOTO 1700
        IF (ABS(Y(3,5)-Y(3,1)).GT.ALIM) GOTO 1700
        IF (ABS(Y(4,5)-Y(4,1)).GT.TLIM) GOTO 1700
        IF (ABS(Y(6,5)-Y(6,1)).LE.TLIM) GOTO 1750
1700    DT=DT/2.
        IF (DT.LT.0.5) THEN
          WRITE(*,1)' BAD EXIT [ DT < 0.5 ]'
          STOP
        ENDIF
        GOTO 800
1750    DTX=DTM
        DO 1800 I=1,7
        IF (Y(1,5)*DTX.LT.ALIM) GOTO 1900
1800    DTX=DTX-2.
1900    CONTINUE
      ENDIF

C     PRE-LAUNCH STABILIZATION CLOCK TIMER
      IF (LAUNCH.EQ.0) THEN
        TT=TT+DT
        DT=DTX
        IF (TT.GE.TS) THEN
          LAUNCH=1
          TT=ANINT(FLY(2,1)-FLY(1,1))
          DAL=ERR*(FLY(2,2)-FLY(1,2))
          IF (DAL.GT.100.) DAL=100.
          CALL VIRON(Y(3,5),BP,DD1,DD3,DD5,LL1,LL3,DD7,DD9,TK,1)
          VB=Y(7,KK)/VT
          CALL MYBLN(VB,GL,GB,GN,DH,SB,DM,RM,HC)
          WV=SW/VS
          GOTO 2100
        ENDIF
        GOTO 2300
      ENDIF

C     ELAPSED FLIGHT TIMER
      T=T+DT

C     INTEGRATION INCREMENT CONTROL TO ENSURE THAT DT IS LESS THAN OR
C     EQUAL TO THE ACTUAL TIME IN CORRESPONDING FLIGHT INCREMENT
      IF (LEAP.EQ.0) THEN
        TT=ANINT(TT-DT)
        IF (TT.LE.DT) THEN
          DT=TT
          LEAP=1
          GOTO 2300
        ENDIF
        GOTO 2300
      ENDIF
```

```
CC     BINARY CHOP ROUTINE TO CHECK ALTITUDE CONVERGENCE AND TO CONTROL
CC     NEW DRAG COEFFICIENT ESTIMATE

C      COMPUTES ALTITUDE CLOSURE
       X=Y(3,5)-FLY(JF+1,2)

C      ALTITUDE NOT WITHIN LIMITS
       IF (ABS(X).GT.DAL) THEN
         IF (X.GT.0.) THEN
           AC3=1.
           SGN=1.
           GOTO 1950
         ENDIF
         AC4=1.
         SGN=-1.
1950     DCD=DCD/(AC3+AC4)
         CD=CD+SGN*DCD
*        RESETS T, TT, DT, LEAP AND RUNGE-KUTTA VARIABLES FOR INTERVAL
*        RECOMPUTATION
         TT=ANINT(FLY(JF+1,1)-FLY(JF,1))
         DT=DTM
         LEAP=0
         T=TTT
         DO 2000 I=1,8
         Y(I,1)=Y(I,6)
2000     Q(I,1)=Q(I,6)
         GOTO 800
       ENDIF

C      ALTITUDE WITHIN LIMITS
       ZFIT=X
       AC3=0.
       AC4=0.
       DCD=.096*CD

CC     END OF ROUTINE

CC     ROUTINE TO COMPUTE AND STORE OUTPUT

       SPEED=ABS((Y(3,5)-XXAL)/(T-TTT))
       VB=Y(7,KK)/VT
       CALL MYBLN(VB,GL,GB,GN,DH,SB,DM,RM,HC)
       CALL VIRON(Y(3,5),BP,DD1,DD3,DD5,LL1,LL3,DD7,DD9,TK,1)
       WV=SW/VS

C      DETERMINES AVERAGE VALUES IN INTERVAL
       XXVB=(XXVB+VB)/2.
       XXHC=(XXHC+HC)/2.
       XXRM=(XXRM+RM)/2.
       XXWV=(XXWV+WV)/2.
       RN=2.*XXRM*XXWV*SPEED
       FRD=(SPEED**2)/(2.*XXRM*G)
```

```
C     STORES AVERAGE VALUES IN INTERVAL
      FR(JF,1)=RN
      FR(JF,2)=FRD
      FR(JF,3)=XXVB
      FR(JF,4)=CD

C     WRITES OUTPUT
      WRITE(5,1)HEAD1
      WRITE(5,7)RN,FRD,XXVB,CD
      WRITE(5,1)HEAD2
      WRITE(5,7)T,Y(3,5),SPEED,ZFIT
      WRITE(5,1)'  '

CC    END OF ROUTINE

C     TERMINATION CHECK
      JF=JF+1
      IF (JF.EQ.INN) GOTO 2500

C     ROUTINE TO SET INITIAL VALUES FOR NEXT ITERATION INTERVAL
      TT=ANINT(FLY(JF+1,1)-FLY(JF,1))
      DAL=ERR*(FLY(JF+1,2)-FLY(JF,2))
      IF (DAL.GT.100.) DAL=100.
      DT=DTM
      LEAP=0
      TTT=T
2100  XXAL=Y(3,5)
      XXHC=HC
      XXRM=RM
      XXVB=VB
      XXWV=WV
      DO 2200 I=1,8
      Y(I,6)=Y(I,5)
2200  Q(I,6)=Q(I,5)
2300  DO 2400 I=1,8
      Y(I,1)=Y(I,5)
2400  Q(I,1)=Q(I,5)
      GOTO 800

C     TERMINATION SEQUENCE: WRITES OUTPUT FILES AND CLOSES FILES
2500  CLOSE(5)
      OPEN(4,FILE=FOUT,STATUS='NEW',FORM='FORMATTED')
      WRITE(4,1)FNAME
      INN=INN-1
      WRITE(4,9)INN
      DO 2550 I=1,INN
2550  WRITE(4,12)FR(I,1),FR(I,2),FR(I,3),FR(I,4)
      CLOSE(4)
      STOP
      END
=================================PROGRAM END ============================

FOR INFORMATION ON SUBROUTINES, SEE PROGRAM FROUDE.

=====================================================================
```

```
      SUBROUTINE PRGR(X0,X1,X2,X3,X4,X5,X6)
      X0=32.174*X1*X2*((X2*X3/X6)**2)*ABS(X4-X5)/X5
      END


      SUBROUTINE MYBLN(VB,GL,GB,GN,DH,SB,DM,RM,HC)
      CALL BRAUN(5,VB,ZB)
      CALL BRAUN(1,ZB,TH)
      CALL BRAUN(2,ZB,RB)
      CALL BRAUN(3,ZB,OO)
      TH=0.01745329252*TH
      GB=1./(1.-OO+RB/SIN(TH))
      GN=GL*GB
      DM=.68312*GN
      RM=DM/2.
      VC=3.1415927*((GN*RB)**3)/3./TAN(TH)
      ZM=.61002*GN
      DH=ZM-ZB*GN
      D=1.-GB
      CALL BRAUN(4,D,OO)
      HC=3.1415927*RM**2
      SB=1.-OO
      END


      SUBROUTINE UPSON(J,X,Y)
      DIMENSION UP(8,3)
      DATA UP/1.235984785,.3809874254,-.048794261,-.374649865,
     *.1451693335,.196113802,-.1170171769,6.,0.,.375469014941,
     *-.0614844739632,.0179374540922,-1.83972293864E-03,
     *2.25926217923E-04,-1.46417871814E-05,6.,.12605508,
     *.061374109851,-8.63907397454E-03,-.090740353,
     *.080087807233,-6.0680228217E-03,-.0101107457963,6./
      Y=UP(1,J)
      III=INT(UP(8,J))
      DO 100 I=1,III
100   Y=Y+UP(I+1,J)*X**I
      END
```

```
      SUBROUTINE NELSON(Y,R,X)
      DIMENSION W(7,4)
      DATA W/-.195390893,-.9529895,.819023889,.581211894,
     *-1.633868387,1.118915175,-.248842045,.19664587,4.64442895,
     *-3.24841125,-3.472893627,7.232507341,-4.25933462,.7859002,
     *3.383282482,-6.630204025,4.076038773,5.56431556,-10.053917,
     *5.4040912,-.859152769,-2.3989589,2.95271P134,-1.645782348,
     *-?.706391858,4.499006036,-2.28485846,.325235255/
      Y=0.
      XX=1.-R
      DO 200 J=1,4
      ZX=W(1,J)
      DO 100 I=2,7
100   ZX=ZX+W(',J)*X**(I-1)
200   Y=Y+ZX*XX**(J-1)
      Y=1.-Y
      END

      SUBROUTINE BRAUN(J,X,Y)
      DIMENSION BR(10,15)
      DATA BR/56.151674,36.869415,-863.384153,4340.60078,-23751.1652,
     *60663.6084,-55452.3401,0.,2.,7.,0.00012929,1.491118,
     *0.89926756885,-12.4916446722,15.757250787,10.1738213142,
     *-26.202606089,0.,2.,7.,.000128,1.791098667,0.901115015,
     *-12.382923389,23.701897323,-8.8057025141,-8.2295252039,
     *0.,?.,7.,-0.00037994265363,0.019166414526,1.87893200657,
     *0.16598678257,1.59554475729,-5.19740094837,2.5981953799,0.,2.,
     *7.,1.,-.0053899018856,-.246383177,0.,0.,0.,0.,.999685,8.,0.,
     *13.47532639,-13.453150437,0.,0.,0.,0.,0.,.998704865,1.,2.,
     *5.70116145,-5.668903443,0.,0.,0.,0.,0.,.996376,1.,2.,
     *3.17395699766,-3.13250803076,0.,0.,0.,0.,0.,.99216,1.,?
     *15.73557415,-67.659164535,90.679194639,-38.706776393,0    .,0.,
     *.96 ,2.,4.,.92472377876,1.83075490637,-13.8588229434,0.,0.,0.,
     *0.,.83035916,8.,0.,.31091805108,.27740682536,-2.63309861357,
     *7.406132484 ,-11.064485699,8.48105265334,-2.69143362928,
     *.488,2.,7.,.340858423189 -.18492521608,0.,0.,0.,0.,0.,.27218,1.,
     *?. .762984275675,-.437916794366,1.02361964267,-1.44464774489,
      .,0.,0.,.0508769,2.,4.,5.3392134074,-28.1122836054,
     *37.0145051127,0.,0.,0.,0.,.00524869,9.,0.,.383179458027,
     *-2.58720408811,0.,0.,0.,0.,0.,.00015,1.,2./
      IF (J.LT.5) GOTO 200
      DO 100 J=5,15
      IF (X.GT.BR(8,J)) GOTO 200
100   CONTINUE
200   LL=INT(BR(9,J))
      MM=INT(BR(10,J))
      IF (LL.EQ.8) THEN
        Y=(-BR(2,J)-SQRT(BR(2,J)*BR(2,J)-4.*BR(3,J)*(BR(1,J)-X)))
        Y=Y/(2.*BR(3,J))
        RETURN
      ENDIF
      Y=BR(1,J)
      DO 300 I=2,MM
```

```
300    Y=Y+BR(I,J)*X**(I-1)
       END
       SUBROUTINE VIRON(XA,PQ,EF,EG,PX,JX,JV,RR,DY,TK,NN)
       COMMON CA,DTI,E(30,2),PE,PO,RO,SW,TI,TIR1,TIRO,TR,TROP,VS
       DATA CC/34.163195/
       X=.3048037*XA
       X=6356.766*X/(6356766.+X)
       IF ((X.LT.E(JX,1)).OR.(X.GE.EF)) THEN
         IF (X.LT.E(JX,1)) THEN
           JV=1
           PX=PO
           GOTO 200
         ENDIF
100      IF (ABS(DY).GE.0.01) PX=PX*((E(JX,2)/EG)**(CC/RR))
         IF (ABS(DY).LT.0.01) PX=PX*EXP(-CC*DX/E(JX,2))
200      JX=JV
         JV=JV+1
         EF=E(JV,1)
         EG=E(JV,2)
         DY=EG-E(JX,2)
         DX=EF-E(JX,1)
         RR=DY/DX
         IF (X.GE.EF) GOTO 100
       ENDIF
       DX=X-E(JX,1)
       TK=E(JX,2)+RR*DX
       IF (ABS(DY).GE.0.01) PQ=PX*((E(JX,2)/TK)**(CC/RR))
       IF (ABS(DY).LT.0.01) PQ=PX*EXP(-CC*DX/E(JX,2))
       IF (NN.EQ.1) THEN
         TR=1.8*TK
         PE=2.08858*PQ
         SW=.0217484*PQ/TK
         RO=SW/32.1741
         VS=7.3023527E-07*(TR**1.5)/(TR+198.72)
         CA=3.30517E-04*(TK**1.5)/(TK+245.4*(10**(-12./TK)))
         TI=TIR1
         IF (X.LT.TROP) TI=TIRO+DTI*(X-E(1,1))
       ENDIF
       END


       SUBROUTINE INFORM
C      3 SEP 1984
       DIMENSION LINE(25)
       CHARACTER*72 LINE
       CHARACTER*15 FINN,FNAME
1      FORMAT(I3)
2      FORMAT(A)
3      FORMAT(4X,A)
4      FORMAT(A,\)
       FINN='QCD.FAX'
       L=0
       OPEN(1,FILE=FINN,FORM='FORMATTED')
```

```
      READ(1,2)FNAME
      READ(1,1)I
      IF (I.NE.678) GO TO 9
C     I: NUMBER OF LINES OF TEXT UNIT TO BE DISPLAYED
6     READ(1,1)I
      IF (I.EQ.0) GO TO 9
      N=0
      IF (I.LE.20) N=(24-I)/2
      CALL CLEAR(25)
      DO 7 K=1,I
7     READ(1,2)LINE(K)
      DO 8 K=1,I
8     WRITE(*,3)LINE(K)
      IF (N.GT.0) CALL CLEAR(N)
      PAUSE
      IF (L.EQ.1 ) GOTO 6
      WRITE(*,4)' DISPLAY PROGRAM NOTES ? [ 0/1 = N/Y ]    '
      READ(*,1)L
      IF (L.EQ.1) GOTO 6
9     CLOSE(1)
      CALL CLEAR(25)
      END
      SUBROUTINE CLEAR(J)
1     FORMAT(A)
      DO 2 I=1,J
2     WRITE(*,1)' '
      END
```

```
      PROGRAM FROUDE
      COMMON CA,DTI,E(30,2),PE,PO,RO,SW,TI,TIR1,TIRO,TR,TROP,VS
      DIMENSION A(5),B(5),C(5),D(8,5),Y(8,5),Q(8,5),RFV(4,20)
      CHARACTER*82 FINN,FOUT,FNAME,FMOD,FXMOD,HEADER
**********************************************************************
*                                                                    *
*   PROGRAM NAME: FROUDE                                20 FEB 1985  *
*                                                                    *
*   THE EXECUTABLE VERSION OF THIS PROGRAM IS DESIGNATED AS 'FLITE'  *
*   WHICH HAS BEEN COMPILED UNDER MICROSOFT FORTRAN77 TO BE RUN ON   *
*   AN IBM PC.                                                       *
*                                                                    *
*   THE DRAG COEFFICIENT AS A FUNCTION OF REYNOLDS NO., FROUDE NO.,  *
*   AND FRACTIONAL VOLUME IS AS FOLLOWS:                             *
*   CD = SUM [RFV(1,J)*((RN**RFV(2,J))*(FR**RFV(3,J))*(VB**RFV(4,J))]  *
*   OVER THE RANGE OF J=1 TO JSEG, WHERE 1.LE.JSEG.LE.20, AND        *
*   WHERE: RN IS THE REYNOLDS NO.                                    *
*          FR IS THE FROUDE NO.                                      *
*          VB IS THE INSTANTANEOUS FRACTIONAL VOLUME OF THE BALLOON  *
*          RFV(I,J) ARE CONSTANTS DETERMINED BY MULTIPLE REGRESSION  *
*                   ANALYSIS OF ACTUAL FLIGHT DATA.                  *
*                                                                    *
*   THE SHAPE OF THE BALLOON IS ASSUMED TO BE THAT OF THE SIMPLIFIED *
*   PARACHUTE-SHAPE MODEL DESCRIBED IN REPORT NO. AFGL-TR-80-0277.   *
*                                                                    *
*   THIS PROGRAM WAS DEVELOPED AT THE AIR FORCE GEOPHYSICS LABORATORY *
*   AS PART OF IN HOUSE WORK UNIT NO. 76591114.                      *
*                                                                    *
**********************************************************************
```

```
      DATA A/3.141592654,.5,.2928932,1.7071068,.1666666666/,GS/96./
      DATA B/0.,2.,1.,1.,2./,C/.01745329252,.5,.2928932,1.7071068,.5/
      DATA BZ/3.6995E-10/,G/32.1741/,RE/20855278./,RA/53.35/,RG/386.076/
      DATA DTM/20./,DTV/0.5/,DT/8./,LL2,LL3/2*1/,LAUNCH/0/
      DATA AC3,AC4,DB,DDO,DD1,T,TT,TTT,VD,VT,VV,Q7,Q8,Q9,Q10,Q11/16*0./
      DATA ALF/1.83E-07/,BET/.682/,GAM/1443./,CV/586.73/,CF/428./
      DATA AYV/.001/,RYV/.114/,TYV/.885/,AYR/.031/,EYR/.031/,RYR/.127/
      DATA TYR/.842/,AYRG/.0028/,WOW/.0048/,CM/.5/,VL/.01/
      DATA CQ1,CQ2,CQ3,CQ4,CQ5,CQ6,CQ7,CQ8,CQ9,CQ10,CQ11/11*1./
      DATA CD,GN1,GN2,GN3/4*1./

      HEADER='        TIME     ALT.  SPEED    RAD        CD       FR       REN
     +        VB'

1     FORMAT(A)
2     FORMAT(1X,'ENTER NAME OF INPUT FILE; B:filespec.FLY     ',\)
3     FORMAT(1X,'AND NAME OF OUTPUT FILE; B:filespec.FLT     ',\)
4     FORMAT(13(E15.8/),E15.8)
5     FORMAT(2F9.3)
6     FORMAT(2F8.0)
7     FORMAT(1X,6E13.7)
8     FORMAT(4(E15.8/),E15.8)
9     FORMAT(I3)
10    FORMAT(10(E15.8/),E15.8)
13    FORMAT(E15.8)
14    FORMAT(1X,3E13.7)
15    FORMAT(3E13.7)
16    FORMAT(1X,F9.1,F8.0,F6.0,F6.1,E9.3,F7.4,,2E9.3)
18    FORMAT(2X,A)

C     CALLS TO THE SCREEN A SERIES OF FACTS AND NOTES REGARDING THIS
C     PROGRAM. THE FILE ADDRESSED CAN BE USER AUGMENTED OR UPDATED.
      CALL INFORM

C     OPENS THE PRINTER AS THE OUTPUT FILE
      OPEN(5,FILE='LPT1',FORM='FORMATTED')

C     OPTIONS: VALUES OF GAS & FILM CONSTANTS
50    WRITE(*,1)' DEFAULT GAS & FILM VALUES ?  [ 0/1 = N/Y ]'
      READ(*,9)I
      IF (I.EQ.0) THEN
        I=1
        WRITE(*,1)' INSERT APPROPRIATE DATA DISK IN B-DRIVE AND'
        WRITE(*,1)' ENTER FILM/GAS FILE NAME, B:filespec.GAF'
        READ(*,1)FNAME
        OPEN(3,FILE=FNAME,FORM='FORMATTED')
        READ(3,4)ALF,BET,GAM,CV,CF,WOW,AYV,RYV,TYV,AYR,EYR,RYR,TYR,AYRG
        CLOSE(3)
        WRITE(5,1)FNAME
      ENDIF
      IF (I.NE.1) GOTO 50
```

```
C     OPTIONS: HEAT TRANSFER COEFFFICIENTS, VIRTUAL MASS COEFFICIENT,
C              AND EFFECTIVE ZERO ASCENT RATE
60    WRITE(*,1)' DEFAULT HEAT XFER COEFs., VIRTUAL MASS COEF. AND'
      WRITE(*,1)' EFFECTIVE ZERO ASCENT RATE ?  [ 0/1 = N/Y ]'
      READ(*,9)I
      IF (I.EQ.0) THEN
        I=1
        WRITE(*,1)' INSERT APPROPRIATE DATA DISK IN B-DRIVE AND'
        WRITE(*,1)' ENTER COEFFICIENT FILE NAME, B:filespec.CMV'
        READ(*,1)FNAME
        OPEN(3,FILE=FNAME,FORM='FORMATTED')
        READ(3,10)CQ1,CQ2,CQ3,CQ4,CQ5,CQ6,CQ7,CQ8,CQ9,CQ10,CQ11
        READ(3,8)GN1,GN2,GN3,CM,VL
        CLOSE(3)
        WRITE(5,1)FNAME
      ENDIF
      IF (I.NE.1) GOTO 60

C     INPUT: FLIGHT DATA
      WRITE(*,1)' INSERT THE FLIGHT DATA DISK IN B-DRIVE AND'
      WRITE(*,2)
      READ(*,1)FINN
      WRITE(*,1)' '
      WRITE(*,3)
      READ(*,1)FOUT
      WRITE(*,1)' '
      OPEN(3,FILE=FINN,FORM='FORMATTED')
      READ(3,1)FNAME
      READ(3,4)SIG,GL,CLEN,AD,VTM,THK,TCP,WB,WP,WT,FL,DBB,SPD,PO
      READ(3,4)DU1,DU2,DU3,DU4,DU5,AGAS,ELL,GH,XD,YG,XL,TS,XIN,XINN
      DO 100 I=1,30
100   READ(3,5)E(I,1),E(I,2)
      CLOSE(3)

C     DRAG MODEL COEFFICIENTS
      WRITE(*,1)' INSERT APPROPRIATE DATA DISK IN B-DRIVE AND'
      WRITE(*,1)' ENTER FILE NAME OF DRAG MODEL: B:filespec.RFV'
      READ(*,1)FMOD
      WRITE(*,1)' '
      OPEN(3,FILE=FMOD,FORM='FORMATTED')
      READ(3,9)FXMOD
      WRITE(*,1)FXMOD
      WRITE(*,1)' '
      READ(3,9)JSEG
      DO 110 J=1,JSEG
      DO 110 I=1,4
110   READ(3,13)RFV(I,J)
      CLOSE (3)
```

```
C     INPUT: INTERVAL GROWTH LIMITS FOR RUNGE-KUTTA VARIABLES
      WRITE(*,1)' ENTER DIFFERENTIAL GROWTH LIMITS: ALT. [ ft ],'
      WRITE(*,1)' ASCENT RATE [ ft/sec ] AND TEMP. [ deg R ]'
      READ(*,15)ALIM,VLIM,TLIM
      WRITE(*,1)' '

C     OPTIONS: MAXIMUM INTEGRATION TIME INCREMENT, INTEGRATION TIME
C              INCREMENT FOR VENTING FROM DUCTS, ENABLEMENT OF ENERGY
C              ABSORPTION BY INFLATANT.
      WRITE(*,1)' TO CHANGE DT [max] OR DT [vent] ENTER NEW NON-ZERO'
      WRITE(*,1)' VALUE, OTHERWISE ZERO. ENTER ZERO OR ONE FOR AGAS.'
      WRITE(*,1)' DTM, DTV, AGAS [ inflatant absorbs, 0/1=N/Y ]'
      WRITE(*,14)DTM,DTV,AGAS
      READ(*,15)GG1,GG2,AGAS
      IF (GG1.GT.0.) DTM=GG1
      IF (GG2.GT.0.) DTV=GG2
      WRITE(*,1)' '

C     OPTION: PARTIAL FLIGHT PROFILE COMPUTATION
      WRITE(*,1)' TO COMPUTE PARTIAL PROFILE,'
      WRITE(*,1)' ENTER: TIME [sec] & ALT. [ft]'
      READ(*,6)TSTOP,ASTOP

C     OUTPUT TO PRINTER
      WRITE(5,18)FOUT
      WRITE(5,1)' '
      WRITE(5,18)FMOD
      WRITE(5,1)' '
      WRITE(5,1)' DTM  DTV  AGAS  ALIM  VLIM  TLIM'
      WRITE(5,7)DTM,DTV,AGAS,ALIM,VLIM,TLIM
      WRITE(5,1)' '

C     PRINTING OF PROFILE HEADINGS
      WRITE(5,1)HEADER
      WRITE(5,1)' '

C     INITIALIZATIONS AND NON-RECURRING COMPUTATIONS
      LGTEN=LOG(10.)
      IN=INT(XIN)
      LL1=IN
      LLO=IN
      EGAS=AGAS
      AYRG=AGAS*AYRG
      AYRG1=1.-AYRG
      AV=AYV*(1.+TYV*AYRG1/(1.-RYV*AYRG1))
      AG=AYRG*TYV/(1.-RYV*AYRG1)
      DBB=DBB/60.
      WTX=0.
      SPD=1.69*SPD
      DD4=PO
      DD5=PO
```

```
C      IR MODEL [ SIMILAR TO REF. 15, PAGE 57 ]
       DO 190 I=2,30
       TROP = E(I-1,1)
       IF (E(I,2).GE.E(I-1,2)) GOTO 200
190    CONTINUE
200    TIRO=1.8*E(1,2)-5.55
       TIR1=.74*TIRO
       DTI=-.26*TIRO/(TROP-E(1,1))

C      ALBEDO MODEL [ SEE REF. 15, FIG. 10 ]
C      NOTE: XL IS IN DEGREES IN THIS MODEL.
       IF (XL.GE.20.) GOTO 300
       RL=-.0025*XL+.15
       GOTO 600
300    IF (XL.GE.30.) GOTO 400
       RL=.1
       GOTO 600
400    IF (XL.GE.40.) GOTO 500
       RL=.005*XL-.05
       GOTO 600
500    RL=.0075*XL-.15

C      CONVERSIONS TO RADIANS
600    XD=C(1)*XD
       XL=C(1)*XL

       CX=COS(XL)*COS(XD)
       SX=SIN(XL)*SIN(XD)

C      APPROXIMATE CAP WEIGHT ROUTINE
       CALL UPSON(1,SIG,ABL)
       GB=CLEN/GL
       CALL NELSON(UO,GB,SIB)
       WC=UO*ABL*WOW*TCF*GL*GL

C      RUNGE-KUTTA ZEROING ROUTINE
       DO 700 J=1,5
       DO 700 I=1,8
       Y(I,J)=0.
700    Q(I,J)=0.

C      ROUTINE TO SET INITIAL VALUES OF R-K VARIABLES
       Y(1,1)=0.
       Y(2,1)=WP+WT
       Y(3,1)=ELL
       CALL VIRON(ELL,BP,DD1,DD3,DD5,LL1,LL3,DD7,DD9,TK,1)
       BE=(1.-.138185)*SW
       VO=(1.+ L)*(WB+WF+WT)/BE
       Y(4,1)=TR
       Y(5,1)=  *FE/TR/RG
       Y(6,1)=
       Y(7,1)=VO
       Y(8,1
```

```
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+                       BEGIN RUNGE-KUTTA ITERATION                   +
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

800   DO 1600 KJ=2,5
      KK=KJ-1
      CALL VIRON(Y(3,KK),BP,DD1,DD3,DD5,LL1,LL3,DD7,DD9,TK,1)
      BE=(1.-.138185*TR/Y(6,KK))*SW

C     ROUTINE TO ADJUST MAXIMUM BALLOON VOLUME DUE TO DEBALLASTING
      IF (WT.NE.WTX) THEN
        X=LOG(1.+WB/(WT+WP))
        CALL UPSON(2,X,SIGX)
        CALL UPSON(3,SIGX,VU)
        VT=VU*GL**3
        WTX=WT
        IF (VT.GT.VTM) VT=VTM
      ENDIF

      IF (KK.EQ.1) THEN

C       FRACTIONAL VOLUME
        VB=Y(7,KK)/VT

C       ROUTINE TO DETERMINE AREA OF CAP ENCLOSING THE GAS BUBBLE
        CALL MYBLN(VB,GL,GB,GN,DH,SB,DM,RM,HC)
        U=U0
        IF (GN.LT.CLEN) CALL NELSON(U,GB,SIG)

C       EFFECTIVE GAS ENVELOPE FILM WEIGHT [ INCLUDING THE CAP PORTION ]
        WE=SB*(WB-WC)+WOW*U*ABL*TCP*GL*GL

C       EFFECTIVE GAS ENVELOPE SURFACE AREA
        SA=SB*ABL*GL*GL

C       EFFECTIVE SURFACE AREA ABSORBING IR [ SEE REF. 15, PAGE 55 ]
        SR=HC
        IF (BE.GE.0.00626) SR=HC+(HC-SA)*LOG(.00626/BE)/LGTEN

      ENDIF

      PG=DH*BE
      SWG=Y(5,KK)/Y(7,KK)
      VSG=ALF*Y(6,KK)**BET
      CG=GAM*VSG
      SPEED=Y(1,KK)

C     PRE LAUNCH WIND SPEED DURING THERMODYNAMIC STABILIZATION
      IF (LAUNCH.EQ.0) SPEED=SPD

      SPDSQ=SPEED*SPEED
      RN=DM*SW*ABS(SPEED)/VS
      BUOY=SW*Y(7,KK)
      WS=Y(2,KK)+Y(5,KK)+WB
      FORCE=BUOY-WS
```

```
C       AERODYNAMIC DRAG ROUTINE FOR RISING BALLOON
        IF (LAUNCH.EQ.1) THEN
          CD=0.
          DO 900 J=2,JSEG
900       CD=CD+REV(1,J)*(RN**REV(2,J))*(FR**REV(3,J))*(VB**REV(4,J))
        ENDIF
        DRAG=.5*RO*CD*HC*SPEED*ABS(SPEED)

        D(1,KJ)=G*(FORCE-DRAG)/(WS+CM*BUOY)

C       PRE-LAUNCH VERTICAL MOTION INHIBITOR
        IF (LAUNCH.EQ.0) D(1,KJ)=0.

        D(2,KJ)=-DB
        D(3,KJ)=Y(1,KK)

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+                           BEGIN HEAT TRANSFER RATES                  +

C       SEE REF. 19, EQ. 24
        EYRG=EGAS*3.42E-06*(Y(6,KK)/1.8)**.8152

        EYRG1=1.-EYRG
        ER=EYR*(1.+TYR*EYRG1/(1.-RYR*EYRG1))
        AR=ER
        EI=EYRG*EYR/(1.-RYR*EYRG1)
        EG=EYRG*TYR/(1.+RYR*EYRG1)

C       NUSSELT NUMBER ROUTINE [ SEE THIS REPORT AND REF. 18 ]
        CALL PRGR(GP,.67,DM,SWG,Y(4,KK),Y(6,KK),VSG)
        IF (GP.LE.15E+07) GNU=2.+.6*GP**0.25
        IF (GP.GT.15E+07) GNU=.13*GP**(1./3.)

C       CONVECTIVE HEAT TRANSFER BETWEEN GAS AND BALLOON ENVELOPE FILM
        Q1=CQ1*GN1*SA*(Y(6,KK)-Y(4,KK))*CG*GNU/DM

        QA=SX+CX*COS(C(1)*(GH-XG+T/240.))

CC      ROUTINE TO DETERMINE EFFECTIVE SOLAR ENERGY

C       SEE REF. 15, EQ. 38
        ARMS=(BP/PO)*(SQRT(1228.6+376750.44*QA*QA)-613.8*ABS(QA))

C       SEE REF. 12, EQ. 48
        TRM1=.5*(EXP(-.65*ARMS)+EXP(-.095*ARMS))

        FV=GS*CS*TRM1
        IF (QA.LT.0.) THEN
          FV=0.
          QB=-SQRT(1.-(RE/(RE+Y(3,KK)))**2)
          IF (QB.LT.QA) THEN

C           OPTICAL AIR MASS ALTITUDE [ SEE REF. 12, EQ. 51 ]
            ZZ=SQRT(1.-QA*QA)*(RE+Y(3,KK))-RE
```

```
C           ATMOSPHERIC PRESSURE FOR OPTICAL AIR MASS ALTITUDE
            CALL VIRON(ZZ,PAM,DDO,DD2,DD4,LLO,LL2,DD6,DD8,TX,2)

            AM=35.1*PAM/PO

C           SEE REF. 12, EQS. 48 & 52
            TRM2=.5*(EXP(-.65*AM)+EXP(-.095*AM))
            FV=GS*CS*TRM2*TRM2/TRM1

          ENDIF
        ENDIF

CC END

C       DIRECT SOLAR ENERGY ABSORPTION
        Q2=CQ2*AV*FV*HC

C       ABSORPTION OF IR ENERGY
        Q3=CQ3*AR*BZ*SR*TI**4

C       NUSSELT NUMBER ROUTINE [ SEE THIS REPORT AND REF. 18 ]
        IF (ABS(SPEED).LT.VL) THEN
          CALL PRGR(GP,.67,DM,SW,Y(4,KK),TR,VS)
          GNU=GN2*(2.+.6*GP**0.25)
        ELSE
          GNU=.37*GN3*RN**0.6
        ENDIF

C       CONVECTIVE HEAT TRANSFER BETWEEN GAS ENVELOPE AND AIR
        Q4=CQ4*SA*(TR-Y(4,KK))*CA*GNU/DM

C       IR ENERGY EMISSION
        Q5=CQ5*ER*BZ*SA*Y(4,KK)**4

C       ROUTINE FOR DETERMINING DIRECTIONAL REFLECTIVITY FACTOR [ SEE
C       REF. 15, FIG. 15 ]
        ZS=57.29578*ATAN(SQRT((1.-QA*QA)/(QA*QA)))
        FF=1.
        IF (ZS.GT.77.2) FF=.09375*ZS-5.4375
        IF ((FF.EQ.1.).AND.(ZS.GT.25.)) FF=.0153*ZS+.6169

C       ABSORPTION OF REFLECTED SOLAR ENERGY
        Q6=CQ6*AV*(2.*HC)*GS*FF*RL*(1.-SQRT(Y(3,KK)/RE/2.))*QA

CC      Q7 THROUGH Q11 ARE BASED ON A MODEL PERMITTING GAS IMPURITIES AND
CC      THUS ENERGY ABSORPTION BY THE INFLATANT [ SEE REFS. 18 & 40 ]

        IF (EI.NE.0.) THEN

C         RADIATIVE EXCHANGE BETWEEN INFLATANT AND ENVELOPE FILM
          Q7=CQ7*EI*BZ*(Y(6,KK)**4-Y(4,KK)**4)*SA

C         ABSORPTION OF DIRECT SOLAR ENERGY
          Q8=CQ8*AG*FV*HC
```

```
C         ABSORPTION OF REFLECTED SOLAR ENERGY
          Q9=CQ9*AG*(2.*HC)*GS*FF*RL*(1.-SQRT(Y(3,KK)/RE/2.))*QA

C         EMISSION OF IR ENERGY BY GAS
          Q10=CQ10*EG*BZ*SA*Y(6,KK)**4

C         ABSORPTION OF IR ENERGY BY GAS
          Q11=CQ11*EG*BZ*SR*TI**4

        ENDIF

CC END

+                              END HEAT TRANSFER RATES                          +
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

        D(4,KJ)=(Q1+Q2+Q3+Q4-Q5+Q6+Q7)/CF/WE

C       ROUTINE FOR EV-13 APEX GAS VALVE OPERATION
        IF ((VV.GT.0.).AND.(KK.EQ.1)) THEN
          IF (PG.GE.0.936) VV=.559-PG/8
          IF (PG.LT.0.936) VV=.72222*SQRT((1.872-PG)*PG)
          VV=6.127*VV*SQRT(SW/SWG-1.)
        ENDIF

        IF(VD.LE.0.) IVENT=0
        IF (IVENT.EQ.0) THEN

C         EQUATIONS FOR FRACTIONAL VOLUME LESS THAN 1
          D(5,KJ)=-VV*SWG
          D(6,KJ)=(Q8+Q9-Q1-Q7-Q10+Q11-SW*Y(7,KK)*D(3,KJ))/Y(5,KK)/(CV+RG)
          D(7,KJ)=Y(7,KK)*(D(5,KJ)/Y(5,KK)+D(6,KJ)/Y(6,KK)+D(3,KJ)/RA/TR)
          D(8,KJ)=-SW*D(3,KJ)

        ELSE

C         EQUATIONS FOR FRACTIONAL VOLUME EQUAL TO 1
          DT=DTV
          VD=AD*SQRT(2.*G*ABS(Y(8,KK)-PE)/SWG)
          D(5,KJ)=-(VV+VD)*SWG
          D(6,KJ)=(Q8+Q9-Q1-Q7-Q10+Q11-PE*D(5,KJ)/SWG)/Y(5,KK)/CV
          D(7,KJ)=0.
          D(8,KJ)=Y(8,KK)*(D(5,KJ)/Y(5,KK)+D(6,KJ)/Y(6,KK))

        ENDIF

        DO 1500 KI=1,8
        QQ=A(KJ)*(D(KI,KJ)-B(KJ)*Q(KI,KK))
        Y(KI,KJ)=Y(KI,KK)+DT*QQ
 1500   Q(KI,KJ)=Q(KI,KK)+3.*QQ-C(KJ)*D(KI,KJ)
 1600   CONTINUE

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+                              END RUNGE-KUTTA ITERATION                        +
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

```
C     ROUTINE TO ADJUST THEORETICAL MAXIMUM BALLOON VOLUME [ WARRANTED
C     ON THE BASIS OF THE ACCURACIES OF ACTUAL CONSTRUCTION PRACTICES ]
C     ANOTHER OPTION WOULD BE TO SET VT=Y(7,5), IVENT=1 AND TO CONTINUE.
      IF (Y(7,5).GT.VT) THEN
        VT=Y(7,1)
        IVENT=1
        GOTO 900
      ENDIF

C     ROUTINE TO DETERMINE IF ANY SELECTED VARIABLE GROWTH RATE IS
C     EXCESSIVE [ NOT APPLICABLE WHEN BALLOON DUCTS ARE VENTING GAS ]
      IF (IVENT.EQ.0) THEN
        IF (ABS(Y(1,5)-Y(1,1)).GT.VLIM) GOTO 1700
        IF (ABS(Y(3,5)-Y(3,1)).GT.ALIM) GOTO 1700
        IF (ABS(Y(4,5)-Y(4,1)).GT.TLIM) GOTO 1700
        IF (ABS(Y(6,5)-Y(6,1)).LE.TLIM) GOTO 1750

C       ROUTINE TO ADJUST INTEGRATION INCREMENT FOR EXCESSIVE GROWTH
1700    DT=DT/2.
        IF (DT.LT.0.5) THEN
          WRITE(*,1)' BAD EXIT [ DT < 0.5 ]'
          STOP
        ENDIF
        GOTO 800

C       ROUTINE TO ADJUST INTEGRATION INCREMENT TO CONTROL ALTITUDE
C       DURING AN ITERATION INTERVAL
1750    DTX=DTM
        DO 1800 I=1,7
        IF (Y(1,5)*DTX.LT.ALIM) GOTO 1900
1800    DTX=DTX-2.
1900    CONTINUE

      ENDIF

C     PRE-LAUNCH STABILIZATION CLOCK TIMER
      IF (LAUNCH.EQ.0) THEN
        TT=TT+DT
        DT=DTX
        IF (TT.GE.TS) THEN
          LAUNCH=1
          DT=1.
          CD=0.
          GOTO 2200
        ENDIF
        GOTO 2300
      ENDIF

C     ELAPSED FLIGHT TIMER
      T=T+DT
```

```
C      ROUTINE TO GENERATE OUTPUT
2200   VD=Y(7,KK)/VT
       CALL MYBLN(VB,GL,GB,GN,DH,SB,DM,RM,HC)
       CALL VIRON(Y(3,KK),BP,DD1,DD3,DD5,LL1,LL3,DD7,DD9,TK,1)
       FR=Y(1,5)/(G*DM)
       RN=DM*SW*ABS(Y(1,5))/VS
       X=60.*Y(1,5)
       WRITE(5,16)T,Y(3,5),X,RM,CD,FR,RN,VB,

C      PARTIAL FLIGHT PROFILE TERMINATION CHECK
       IF((T.GE.TSTOP).OR.(Y(3,5).GE.ASTOP)) GOTO 2500

C      INTEGRATION INCREMENT CONTROL DURING FIRST 10 SECONDS OF FLIGHT
       IF (T.GT.10.) DT=DTX

C      ROUTINE TO RE-INITIALIZE RUNGE-KUTTA ITERATION
2300   DO 2400 I=1,8
       Y(I,1)=Y(I,5)
2400   Q(I,1)=Q(I,5)
       GOTO 800

2500   CLOSE(5)
       STOP

       END
```

```
      SUBROUTINE PRGR(X0,X1,X2,X3,X4,X5,X6)

***********************************************************************
*                                                                     *
*     COMPUTES THE PRODUCT OF THE GRASHOF NO. AND THE PRANDTL NO.     *
*                                                                     *
***********************************************************************

      X0=32.174*X1*X2*((X2*X3/X6)**2)*ABS(X4-X5)/X5

      END
```

```
      SUBROUTINE NELSON(Y,R,X)

***********************************************************************
*                                                                     *
*   COMPUTES FRACTIONAL CAP AREA [ IN FULL OR IN PART ] FOR A GIVEN   *
*   SIGMA AND LENGTH [ AS A FRACTION OF THE ACTUAL GORELENGTH ] AS    *
*   MEASURED FROM THE THEORETICAL APEX POSITION                       *
*                                                                     *
***********************************************************************

      DIMENSION W(7,4)

      DATA W/-.195390893,-.9529895,.819023889,.581211894,
     *-1.633860387,1.118915175,-.248842045,.19664587,4.64442895,
     *-3.24041125,-3.472893627,7.232507341,-4.25933462,.7859002,
     *3.383282482,-6.630204025,4.076030773,5.56431556,-10.053917,
     *5.4040912,-.859152769,-2.3989589,2.952718136,-1.645782348,
     *-2.706371858,4.499006036,-2.28485846,.325235255/

      Y=0.
      XX=1.-R
      DO 200 J=1,4
      ZX=W(1,J)
      DO 100 I=2,7
100   ZX=ZX+W(I,J)*X**(I-1)
200   Y=Y+ZX*XX**(J-1)
      Y=1.-Y

      END
```

```
      SUBROUTINE MYBLN(VB,GL,GB,GN,DH,SB,DM,RM,HC)

***********************************************************************
*                                                                     *
*   COMPUTES GEOMETRICAL CHARACTERISTICS OF THE SIMPLIFIED PARACHUTE  *
*   SHAPE BALLOON                                                     *
*                                                                     *
***********************************************************************

C     COMPUTES DIMENSIONLESS HEIGHT COORDINATE (ZB) OF THE GENERATOR
C     SHAPE [ AT LINE OF INTERSECTION WITH THE BASE CONE ] AS A FUNCTION
C     OF FRACTIONAL VOLUME (VB)
      CALL BRAUN(5,VB,ZB)

C     COMPUTES HALF-ANGLE (TH) AT NADIR AS A FUNCTION OF ZB
      CALL BRAUN(1,ZB,TH)

C     COMPUTES DIMENSIONLESS HORIZONTAL RADIUS COORDINATE (RB) OF THE
C     GENERATOR SHAPE [ AT HEIGHT ZB ] AS A FUNCTION OF ZB
      CALL BRAUN(2,ZB,RB)

C     COMPUTES DIMENSIONLESS GORELENGTH COORDINATE (OO) OF THE GENERATOR
C     SHAPE [ AT HEIGHT ZB ] AS A FUNCTION OF ZB
      CALL BRAUN(3,ZB,OO)

C     CONVERTS TH FROM DEGREES TO RADIANS
      TH=0.01745329252*TH

C     COMPUTES GB, THE RATIO OF THE GENERATOR SHAPE GORELENGTH TO THE
C     ACTUAL BALLOON GORELENGTH
      GB=1./(1.-OO+RB/SIN(TH))

C     COMPUTES THE GENERATOR SHAPE GORELENGTH
      GN=GL*GB

C     COMPUTES THE MAXIMUM HORIZONTAL DIAMETER OF THE GENERATOR SHAPE
C     [ THE DIAMETER OF THE IDEALIZED, PARTIALLY FULL BALLOON ]
      DM=.68312*GN

C     CONVERTS THE DIAMETER TO THE RADIUS
      RM=DM/2.

C     COMPUTES THE VOLUME OF THE TANGENT CONE
      VC=3.1415927*((GN*RB)**3)/3./TAN(TH)

C     COMPUTES THE HEIGHT OF THE GENERATOR SHAPE
      ZM=.61002*GN

C     COMPUTES THE [illegible]IGHT OF THE HELIUM FILLED [illegible]UME
      DH=ZM-ZB*GN

C     COMPUTES DIMENSION[illegible]IFFERENCE BETWEEN THE ACTUAL BALLOON
C     GORELENGTH AND THE I[illegible]NTANEOUS GENERATOR SHAPE GORELENGTH
      [illegible]=1.[illegible]B
```

```
C     COMPUTES FRACTIONAL SURFACE AREA OF ACTUAL BALLOON BELOW THE
C     LOCUS OF POINTS CORRESPONDING TO DIMENSIONLESS LOCATION (O)
      CALL BRAUN(4,O,OO)

C     COMPUTES FRACTIONAL SURFACE AREA OF THE ACTUAL BALLOON SHELL
C     ASSUMED TO APPROXIMATE THE AREA OF THE ENVELOPE SURROUNDING
C     THE GAS BUBBLE
      SB=1.-OO

C     COMPUTES AREA OF THE MAXIMUM HORIZONTAL CROSSSECTION OF THE
C     BALLOON
      HC=3.1415927*RM**2

      END

=========================================================================

      SUBROUTINE UPSON(J,X,Y)

*************************************************************************
*                                                                       *
*   ROUTINE TO COMPUTE CERTAIN CHARACTERISTICS OF THE UPSON NATURAL     *
*   SHAPE: BASED ON TABULAR DATA ORIGINALLY PREPARED BY J.H.SMALLEY     *
*                                                                       *
*   FOR J=1, OUTPUTS NON DIMENSIONAL SURFACE AREA                       *
*            A/S**2 = FCN(SIGMA), R**2 = .999999711                     *
*                                                                       *
*   FOR J=2  OUTPUTS SIGMA                                              *
*            SIGMA = FCN(G/L), R**2 = .999999998                        *
*                                                                       *
*   FOR J=3, OUTPUTS NON DIMENSIONAL VOLUME                             *
*            V/S**3 = FCN(SIGMA), R**2 = .999999324                     *
*                                                                       *
*************************************************************************

      DIMENSION UP(8,3)

      DATA UP/1.235984785,.38898742254,-.048794261,-.374649865,
     *.1451693335,.1961138012,-.1170171769,6.,0.,.375469014941,
     *-.061484472963 2,.0179374540922,-1.83972293864E-03,
     *2.25926217923E-04,-1.46417871814E-05,6.,.12605508,
     *.061374109851,-8.63907397454E-03,-.090740353,
     *.080087807233,-6.06802282 17E-03,-.0101107457963,6./

      Y=UP(1,J)
      III=INT(UP(8,J))
      DO 100 I=1,III
100   Y=Y+UP(I+1,J)*Y**I

      END
```

```
      SUBROUTINE BRAUN(J,X,Y)

*******************************************************************
*                                                                 *
*   MATHEMATICAL MODELS OF FUNCTIONS USED IN THE SUBROUTINE MYBLN *
*                                                                 *
*******************************************************************

      DIMENSION BR(10,15)

      DATA BR/56.151674,36.869415,-863.384153,4340.60078,-23751.1652,
     *60663.6084,-55452.3401,0.,2.,7.,0.00012929,1.491118,
     *0.89926756885,-12.4916446722,15.759250787,10.1738213142,
     *-26.202606089,0.,2.,7.,.000128,1.791098667,0.901115015,
     *-12.382923389,23.701897323,-8.8057025141,-8.2295252039,
     *0.,2.,7.,-0.00037994265363,0.019166414526,1.87893200657,
     *0.10698678257,1.59554475729,-5.19740094837,2.5981953799,0.,2.,
     *7.,1.,-.0053899018856,-.246383177,0.,0.,0.,0.,.999685,8.,0.,
     *13.47532639,-13.453150437,0.,0.,0.,0.,0.,.998704865,1.,2.,
     *5.70116145,-5.668903443,0.,0.,0.,0.,0.,.996376,1.,2.,
     *3.17395699766,-3.13250803076,0.,0.,0.,0.,0.,.99216,1.,2.,
     *15.73557415,-67.659164535,90.679194639,-38.706776393,0.,0.,0.,
     *.963,2.,4.,.92472377876,1.83075490637,-13.8588229434,0.,0.,0.,
     *0.,.83035916,8.,0.,.31091805108,.27740682536,-2.63309861357,
     *7.406132484 ,-11.064485699,8.48105265334,-2.69143362928,
     *.488,2.,7.,.340858423189,-.18492521608,0.,0.,0.,0.,0.,.27218,1.,
     *2.,.36298?275675,-.437916794366,1.02361964267,-1.44463774489,
     *0.,0.,0.,.0508769,2.,4.,5.3392134074,-28.1122836054,
     *37.0145051127,0.,0.,0.,0.,.00524869,8.,0.,.383179458027,
     *-2.58720408811.0.,0.,0..0.,0.,.00015,1.,2./

       IF (J.LT.5) GOTO 200
       DO 100 J=5,15
       IF (X.GT.BR(8,J)) GOTO 200
  100  CONTINUE
  200  LL=INT(BR(9,J))
       MM=INT(BR(10,J))
       IF (LL.EQ.8) THEN
         Y=(-BR(2,J)-SQRT(BR(2,J)*BR(2,J)-4.*BR(3,J)*(BR(1,J)-X)))
         Y=Y/(2.*BR(3,J))
         RETURN
       ENDIF
       Y=BR(1,J)
       DO 300 I=2,MM
  300  Y=Y+BR(I,J)*X**(I-1)

       END
```

```
      SUBROUTINE VIRON(XA,PQ,EF,EG,PX,JX,JV,RR,DY,TK,NN)

************************************************************************
*                                                                      *
*     ATMOSPHERIC MODEL BASED ON 1962 U.S. STANDARD ATMOSPHERE;        *
*     WHEN NN = 2, COMPUTES ONLY PRESSURE FOR AIR MASS ALTITUDE        *
*                                                                      *
************************************************************************

      COMMON CA,DTI,E(30,2),PE,PO,RO,SW,TI,TIR1,TIRO,TR,TROP,VS

      DATA CC/34.163195/

      X=.3048037*XA
      X=6356.766*X/(6356766.+X)
      IF ((X.LT.E(JX,1)).OR.(X.GE.EF)) THEN
        IF (X.LT.E(JX,1)) THEN
          JV=1
          PX=PO
          GOTO 200
        ENDIF
100     IF (ABS(DY).GE.0.01) PX=PX*((E(JX,2)/EG)**(CC/RR))
        IF (ABS(DY).LT.0.01) PX=PX*EXP(-CC*DX/E(JX,2))
200     JX=JV
        JV=JV+1
        EF=E(JV,1)
        EG=E(JV,2)
        DY=EG-E(JX,2)
        DX=EF-E(JX,1)
        RR=DY/DX
        IF (X.GE.EF) GOTO 100
      ENDIF
      DX=X-E(JX,1)
      TK=E(JX,2)+RR*DX
      IF (ABS(DY).GE.0.01) PQ=PX*((E(JX,2)/TK)**(CC/RR))
      IF (ABS(DY).LT.0.01) PQ=PX*EXP(-CC*DX/E(JX,2))
      IF (NN.EQ.1) THEN

C         TEMPERATURE, RANKINE
          TR=1.8*TK

C         ATMOSPHERIC PRESSURE, LBS/FT**2
          PE=2.08858*PQ

C         SPECIFIC WEIGHT OF AIR, LBS/FT**3
          SW=.0217484*PQ/TK

C         AIR DENSITY, SLUGS/FT**3
          RO=SW/32.1741

C         VISCOSITY OF AIR, LBS/FT/SEC
          VS=7.3023527E-07*(TR**1.5)/(TR+198.72)
```

```
C        THERMAL CONDUCTIVITY OF AIR, LBS/SEC/DEGREE RANKINE
         CA=3.30517E-04*(TK**1.5)/(TK+245.4*(10**(-12./TK)))

C        RADIATION TEMPERATURE OF AIR, RANKINE
         TI=TIR1
         IF  (X.LT.TROP) TI=TIRO+DTI*(X-E(1,1))

       ENDIF

       END

=====================================================================

       SUBROUTINE INFORM

**********************************************************************
*                                                                    *
*        ACCESSES PROGRAM NOTES [ AUTHOR'S STANDARD ROUTINE ]        *
*                                                                    *
**********************************************************************

       DIMENSION LINE(25)
       CHARACTER*72 LINE
       CHARACTER*15 FINN,FNAME
1      FORMAT(I3)
2      FORMAT(A)
3      FORMAT(4X,A)
4      FORMAT(A,\)
       FINN='FLITE.FAX'
       L=0
       OPEN(1,FILE=FINN,FORM='FORMATTED')
       READ(1,2)FNAME
       READ(1,1)I
       IF (I.NE.678) GO TO 9
C      I: NUMBER OF LINES OF TEXT UNIT TO BE DISPLAYED
6      READ(1,1)I
       IF (I.EQ.0) GO TO 9
       N=0
       IF (I.LE.20) N=(24-I)/2
       CALL CLEAR(25)
       DO 7 K=1,I
7      READ(1,2)LINE(K)
       DO 8 K=1,I
8      WRITE(*,3)LINE(K)
       IF (N.GT.0) CALL CLEAR(N)
       PAUSE
       IF (L.EQ.1 ) GOTO 6
       WRITE(*,4)' DISPLAY PROGRAM NOTES ? [ 0/1 = N/Y ]     '
       READ(*,1)L
       IF (L.EQ.1) GOTO 6
9      CLOSE(1)
       CALL CLEAR(25)
       END
```

```
      SUBROUTINE CLEAR(J)

***********************************************************************
*                                                                     *
*    CALLED ONLY BY SUBROUTINE INFORM TO CLEAR SCREEN OF PREVIOUS     *
*    NOTE [ AUTHOR'S STANDARD ROUTINE ]                               *
*                                                                     *
***********************************************************************

1     FORMAT(A)
      DO 2 I=1,J
2     WRITE(*,1)' '
      END

=========================== END OF PROGRAM ============================
=======================================================================
```

```
      PROGRAM CDMODEL

************************************************************************
*                                                                      *
*     PROGRAM: CDMODEL                              30 APRIL 1985      *
*                                                                      *
*     THE EXECUTABLE VERSION OF THIS PROGRAM IS DESIGNATED AS 'CD'     *
*     WHICH HAS BEEN COMPILED UNDER MICROSOFT FORTRAN 77 TO BE RUN     *
*     ON AN IBM PC.                                                    *
*                                                                      *
*     THIS PROGRAM FORMATS DRAG COEFFICIENT MODELS FOR USE WITH THE    *
*     EXECUTABLE PROGRAM 'FLITE'.                                      *
*                                                                      *
*     THIS PROGRAM WAS DEVELOPED AT AND FOR THE AIR FORCE GEOPHYSICS   *
*     LABORATORY AS PART OF IN-HOUSE WORK UNIT NO. 76591114.           *
*                                                                      *
************************************************************************

      DIMENSION RFV(4,20)

      CHARACTER*15 FMOD

2     FORMAT(A,'  ',\)
3     FORMAT(I3)
11    FORMAT(A)
13    FORMAT(E15.8)
19    FORMAT(1X,' COEF. FOR TERM ',I3,':',\)
20    FORMAT(1X,' REYNOLDS NUMBER EXPONENT ',':   ',\)
21    FORMAT(1X,' FROUDE NUMBER EXPONENT ',':     ',\)
22    FORMAT(1X,' FRACTIONAL VOLUME EXPONENT ',':',\)

      CALL INFORM

100   WRITE(*,2)' ENTER NUMBER OF TERMS IN MODEL: [ .LE.20 ]'
      READ(*,3)JSEG
      IF (JSEG.GT.20.OR.JSEG.LT.1) GOTO 100
      WRITE(*,11)' '
      DO 110 J=1,JSEG
      WRITE(*,19)J
      READ(*,13)RFV(1,J)
      WRITE(*,11)' '
      WRITE(*,20)
      READ(*,13)RFV(2,J)
      WRITE(*,21)
      READ(*,13)RFV(3,J)
      WRITE(*,22)
      READ(*,13)RFV(4,J)
110   WRITE(*,11)' '

      WRITE(*,11)' INSERT PROPER DISK IN B-DRIVE, AND'
      WRITE(*,2)' ENTER NEW MODEL NAME: B:filespec.RFV'
      READ(*,11)FMOD
      OPEN(3,FILE=FMOD,STATUS='NEW',FORM='FORMATTED')
      WRITE(3,11)FMOD
      WRITE(3,3)JSEG
```

```
      DO 120 J=1,JSEG
      DO 120 I=1,4
120   WRITE(3,13)RFV(I,J)
      CLOSE (3)

      END

      SUBROUTINE INFORM
C     3 SEP 1984
      DIMENSION LINE(25)
      CHARACTER*72 LINE
      CHARACTER*15 FINN,FNAME
1     FORMAT(I3)
2     FORMAT(A)
3     FORMAT(4X,A)
4     FORMAT(A,\)
      FINN='CD.FAX'
      L=0
      OPEN(1,FILE=FINN,FORM='FORMATTED')
      READ(1,2)FNAME
      READ(1,1)I
      IF (I.NE.678) GO TO 9
C     I: NUMBER OF LINES OF TEXT UNIT TO BE DISPLAYED
6     READ(1,1)I
      IF (I.EQ.0) GO TO 9
      N=0
      IF (I.LE.20) N=(24-I)/2
      CALL CLEAR(25)
      DO 7 K=1,I
7     READ(1,2)LINE(K)
      DO 8 K=1,I
8     WRITE(*,3)LINE(K)
      IF (N.GT.0) CALL CLEAR(N)
      PAUSE
      IF (L.EQ.1 ) GOTO 6
      WRITE(*,4)' DISPLAY PROGRAM NOTES ? [ 0/1 = N/Y ]     '
      READ(*,1)L
      IF (L.EQ.1) GOTO 6
9     CLOSE(1)
      CALL CLEAR(25)
      END

      SUBROUTINE CLEAR(J)
1     FORMAT(A)
      DO 2 I=1,J
2     WRITE(*,1)' '
      END
```

```
      PROGRAM DATAFORM

C     THIS IS A FORTRAN PROGRAM TO FORMAT INPUT DATA FOR 'QCD.EXE'
C     AND FOR 'FLITE.EXE'.

      DIMENSION X(28),E(9,2),Y(30,2),Z(100,4)

      CHARACTER*64 FINN,FMAT

      DATA E/0.,11.,2  ,32.,47.,52.,61.,79.,88.7,288.15,216.65,216.65,
     *228.65,270.65,27 .65,252.65,180.65,180.65/,IQ/0/,AIQ/1./
      DATA AC3,AC4,E1/3*0./,CC/34.163195/,JD,JIFF/8,1/,H1/1013.25/

1     FORMAT('  ENTER NAME OF OUTPUT FILE; B:filespec.FLY          ',\)
2     FORMAT(A,'  ',\)
3     FORMAT(I3)
4     FORMAT(1X,F8.0,1X,F8.0,1X,E15.8)
5     FORMAT(1X,F8.0,1X,F8.0)
6     FORMAT('  NEW FLIGHT DATA FILE [N/Y = 0/1] ?  ',\)
7     FORMAT('  NEW GAS/FILM COEFFICIENT FILE [N/Y = 0/1] ?  ',\)
8     FORMAT('  NEW HEAT XFER COEFFICIENT FILE [N/Y = 0/1] ?  ',\)
9     FORMAT('  ENTER NAME OF OUTPUT FILE; B:filespec.GAF          ',\)
10    FORMAT('  ENTER NAME OF OUTPUT FILE; B:filespec.CMV          ',\)
11    FORMAT(A)
12    FORMAT('  IS INPUT FOR PROGRAM "FROUDE" [N/Y = 0/1] ?  ',\)

      CALL INFORM

C     CREATES B:filespec.FLY
      WRITE(*,6)
      READ(*,3)IAM
      IF (IAM.EQ.1) THEN
        WRITE(*,1)
        READ(*,11)FINN
        FMAT='(E15.8)'
50      WRITE(*,12)
        READ(*,3)JFD
        IF((JFD.NE.0).OR.(JFD.NE.1)) GOTO 50
        WRITE(*,2)'  SIGMA'
        READ(*,FMAT)X(1)
        WRITE(*,2)'  GORELENGTH [FT]'
        READ(*,FMAT)X(2)
        WRITE(*,2)'  CAP LENGTH [FT]'
        READ(*,FMAT)X(3)
        WRITE(*,2)'  TOTAL DUCT AREA [SQ FT]'
        READ(*,FMAT)X(4)
        WRITE(*,2)'  MAXIMUM BALLOON VOLUME [CU FT]'
        READ(*,FMAT)X(5)
        WRITE(*,2)'  FILM THICKNESS [mils]'
        READ(*,FMAT)X(6)
        WRITE(*,2)'  CAP THICKNESS [mils]'
        READ(*,FMAT)X(7)
        WRITE(*,2)'  BALLOON WEIGHT [LBS]'
        READ(*,FMAT)X(8)
        WRITE(*,2)'  IRREDUCIBLE PAYLOAD WEIGHT [LBS]'
```

```
      READ(*,FMAT)X(9)
      WRITE(*,2)'  BALLAST WEIGHT [LBS]'
      READ(*,FMAT)X(10)
      WRITE(*,2)'  FRACTIONAL FREE LIFT'
      READ(*,FMAT)X(11)
      WRITE(*,2)'  BALLAST POUR RATE [LBS/MIN]'
      READ(*,FMAT)X(12)
      WRITE(*,2)'  GROUND WIND SPEED [knots]'
      READ(*,FMAT)X(13)
      WRITE(*,2)'  LAUNCH SITE ATM. PRESSURE [mbs]'
      READ(*,FMAT)X(14)
      X(15)=1.
      X(16)=1.
      X(17)=1.
      X(18)=1.
      X(19)=1.
      IF (JFD.EQ.0) THEN
        WRITE(*,2)'  ESTIMATED DRAG COEFFICIENT [0.8]'
        READ(*,FMAT)X(15)
        WRITE(*,2)'  MAX. REL. ERR. IN ALTITUDE CLOSURE [0.01]'
        READ(*,FMAT)X(16)
        WRITE(*,2)'  DRAG COEFFICIENT ADJUSTMENT [CD/4]'
        READ(*,FMAT)X(18)
        WRITE(*,2)'  ENABLE GAS ABSORPTION [N/Y = 0./1.]'
        READ(*,FMAT)X(20)
      ENDIF
      WRITE(*,2)'  LAUNCH SITE ELEVATION [FT]'
      READ(*,FMAT)X(21)
      WRITE(*,2)'  GREENWICH HOUR ANGLE [degrees]'
      READ(*,FMAT)X(22)
      WRITE(*,2)'  DECLINATION [degrees]'
      READ(*,FMAT)X(23)
      WRITE(*,2)'  LONGITUDE [degrees]'
      READ(*,FMAT)X(24)
      WRITE(*,2)'  LATITUDE [degrees]'
      READ(*,FMAT)X(25)
      WRITE(*,2)'  TEMPERATURE STABILIZATION TIME [sec]'
      READ(*,FMAT)X(26)
      WRITE(*,2)'  NUMBER OF LOCAL ATMOSPHERE POINTS [30 max.]'
      READ(*,FMAT)X(27)
      WRITE(*,2)'  NUMBER OF FLIGHT PROFILE POINTS [100 max]'
      READ(*,FMAT)X(28)

      IN=INT(X(27))
      INN=INT(X(28))

      DO 100 I=1,30
      Y(I,1)=0.
100   Y(I,2)=0.

      DO 200 I=1,INN
      Z(I,1)=0.
      Z(I,2)=0.
      Z(I,3)=0.
200   Z(I,4)=0.
```

```
C       INPUT LOCAL ALTITUDE-TEMPERATURE PROFILE
        WRITE(*,11)'  LOCAL ATMOSPHERE: ALT. [GKm] - TEMP. [Kelvin]'
        WRITE(*,2)'  IF RANKINE ENTER 1.8; IF KELVIN ENTER 1.0'
        READ(*,FMAT)TCON
        FMAT='(F9.3)'
        DO 300 I=1,IN
        WRITE(*,2)'  ALTITUDE [GKm]'
        READ(*,FMAT)Y(I,1)
        WRITE(*,2)'  ABSOLUTE TEMPERATURE'
        READ(*,FMAT)TABS
300     Y(I,2)=TABS/TCON

        WRITE(*,11)'  FLIGHT PROFILE: ALT. [ft] & ELAPSED TIME [sec]'
350     WRITE(*,2)'  ENTER ALT. SOURCE NO.: 1 = PRESSURE, 2 = RADAR ?'
        READ(*,3)ITEM
        IF ((ITEM.LT.1).OR.(ITEM.GT.2)) GOTO 350

C       COMPUTES STD. ATM. PRESSURE FOR INDICATED FLIGHT ALTITUDE
        FMAT='(F8.0)'
        DO 600 I=1,INN
        WRITE(*,2)'  ALTITUDE [ft]'
        READ(*,FMAT)Z(I,2)
        WRITE(*,2)'  TIME [sec]'
        READ(*,FMAT)Z(I,1)
        IF (ITEM.EQ.1) THEN
          Q=.3048037*Z(I,2)
          Q=6356.766*Q/(6356766.+Q)
          IF ((Q.GE.E(JD,1)).AND.(Q.LT.E1)) GOTO 500
400       JD=JIFF
          JIFF=JIFF+1
          E1=E(JIFF,1)
          E2=E(JIFF,2)
          TD=E2-E(JD,2)
          DX=E1-E(JD,1)
          R=TD/DX
          IF (Q.LT.E1) GOTO 500
          IF (ABS(TD).GE.0.01) H1=H1*((E(JD,2)/E2)**(CC/R))
          IF (ABS(TD).LT.0.01) H1=H1*EXP(-CC*DX/E(JD,2))
          GOTO 400
500       DX=Q-E(JD,1)
          TK=E(JD,2)+R*DX
          IF (ABS(TD).GE.0.01) Z(I,3)=H1*((E(JD,2)/TK)**(CC/R))
          IF (ABS(TD).LT.0.01) Z(I,3)=H1*EXP(-CC*DX/E(JD,2))
        ENDIF
600     WRITE(*,4)Z(I,1),Z(I,2),Z(I,3)

        IF (ITEM.EQ.2) GOTO 2100

        E1=-5000.
        JD=IN

C       CORRECTS ALTITUDE FOR LOCAL ATMOSPHERIC PRESSURE
        DO 2000 I=1,INN
        AL=Z(I,2)
```

```
      DAL=800.
      DZ3=0.005*Z(I,3)
700   Q=.3048037*AL
      Q=6356.766*Q/(6356766.+Q)
      IF ((Q.GE.Y(JD,1)).AND.(Q.LT.E1)) GOTO 900
      IF (Q.GE.E1) GOTO 800
      JIFF=1
      H1=X(14)
800   JD=JIFF
      JIFF=JIFF+1
      E1=Y(JIFF,1)
      E2=Y(JIFF,2)
      TD=E2-Y(JD,2)
      DX=E1-Y(JD,1)
      R=TD/DX
      IF (Q.LT.E1) GOTO 900
      IF (ABS(TD).GE.0.01) H1=H1*((Y(JD,2)/E2)**(CC/R))
      IF (ABS(TD).LT.0.01) H1=H1*EXP(-CC*DX/Y(JD,2))
      GOTO 800
900   DX=Q-Y(JD,1)
      TK=Y(JD,2)+R*DX
      IF (ABS(TD).GE.0.01) BP=H1*((Y(JD,2)/TK)**(CC/R))
      IF (ABS(TD).LT.0.01) BP=H1*EXP(-CC*DX/Y(JD,2))

C     PRESSURE CONVERGENCE CHECK
      Q=AIQ*(BP-Z(I,3))

C     CORRECTS MONOTONIC ASSUMPTION IF, REQUIRED
      IF (IQ.LT.2) THEN
        IQ=IQ+1
        IF (IQ.EQ.2) THEN
          IF (((Q*QQQ).GT.0.).AND.(ABS(QQQ).GT.ABS(Q))) THEN
            AIQ=-1.
            AC3=0.
            AC4=0.
            AL=Z(I,2)
            DAL=800.
            GOTO 700
          ENDIF
          GOTO 950
        ENDIF
        QQQ=Q
      ENDIF

C     LIMIT CHECK
950   IF (ABS(Q).GT.DZ3) THEN

C       NOT WITHIN LIMITS
        IF (Q.LT.0.) GOTO 1000
        AC3=1.
        SGN=1.
        GOTO 1100
1000    AC4=1.
        SGN=-1.
1100    DAL=DAL/(AC3+AC4)
```

```
      AL=AL+SGN*DAL
      GOTO 700
    ENDIF
C   WITHIN LIMITS
    AC3=0.
    AC4=0.
    Z(I,4)=Z(I,2)
    Z(I,2)=AL
2000 WRITE(*,5)Z(I,2),Z(I,4)
2100 OPEN(3,FILE=FINN,STATUS='NEW',FORM='FORMATTED')
    WRITE(3,11)FINN
    FMAT='(E15.8)'
    DO 2200 I=1,28
2200 WRITE(3,FMAT)X(I)
    FMAT='(2F9.3)'
    DO 2300 I=1,30
2300 WRITE(3,FMAT)Y(I,1),Y(I,2)
    FMAT='(2F8.0)'
    DO 2400 I=1,INN
2400 WRITE(3,FMAT)Z(I,1),Z(I,2)
    CLOSE(3)
   ENDIF
C  CREATES B:filespec.GAF
   WRITE(*,7)
   READ(*,3)IAM
   IF (IAM.EQ.1) THEN
     WRITE(*,9)
     READ(*,11)FINN
     FMAT='(E15.8)'
     WRITE(*,2)'  ALF'
     READ(*,FMAT)X(1)
     WRITE(*,2)'  BET'
     READ(*,FMAT)X(2)
     WRITE(*,2)'  GAM'
     READ(*,FMAT)X(3)
     WRITE(*,2)'  CV'
     READ(*,FMAT)X(4)
     WRITE(*,2)'  CF'
     READ(*,FMAT)X(5)
     WRITE(*,2)'  WOW'
     READ(*,FMAT)X(6)
     WRITE(*,2)'  AYV'
     READ(*,FMAT)X(7)
     WRITE(*,2)'  RYV'
     READ(*,FMAT)X(8)
     WRITE(*,2)'  TYV'
     READ(*,FMAT)X(9)
     WRITE(*,2)'  AYR'
     READ(*,FMAT)X(10)
     WRITE(*,2)'  EYR'
     READ(*,FMAT)X(11)
```

```
      WRITE(*,2)   RYR'
      READ(*,FMAT)X(12)
      WRITE(*,2)'  TYR'
      READ(*,FMAT)X(13)
      WRITE(*,2)'  AYRG'
      READ(*,FMAT)X(14)
      OPEN(3,FILE=FINN,STATUS='NEW',FORM='FORMATTED')
      WRITE(3,11)FINN
      DO 2600 I=1,14
2600  WRITE(3,FMAT)X(I)
      CLOSE(3)
    ENDIF

C   CREATES B:filespec.CMV
    WRITE(*,8)
    READ(*,3)IAM
    IF (IAM.EQ.1) THEN
      WRITE(*,10)
      READ(*,11)FINN
      FMAT='(E15.8)'
      WRITE(*,2)'  CQ1'
      READ(*,FMAT)X(1)
      WRITE(*,2)'  CQ2'
      READ(*,FMAT)X(2)
      WRITE(*,2)'  CQ3'
      READ(*,FMAT)X(3)
      WRITE(*,2)'  CQ4'
      READ(*,FMAT)X(4)
      WRITE(*,2)'  CQ5'
      READ(*,FMAT)X(5)
      WRITE(*,2)'  CQ6'
      READ(*,FMAT)X(6)
      WRITE(*,2)'  CQ7'
      READ(*,FMAT)X(7)
      WRITE(*,2)'  CQ8'
      READ(*,FMAT)X(8)
      WRITE(*,2)'  CQ9'
      READ(*,FMAT)X(9)
      WRITE(*,2)'  CQ10'
      READ(*,FMAT)X(10)
      WRITE(*,2)'  CQ11'
      READ(*,FMAT)X(11)
      WRITE(*,2)'  GN1'
      READ(*,FMAT)X(12)
      WRITE(*,2)'  GN2'
      READ(*,FMAT)X(13)
      WRITE(*,2)'  GN3'
      READ(*,FMAT)X(14)
      WRITE(*,2)'  VIRTUAL MASS COEFFICIENT'
      READ(*,FMAT)X(15)
      WRITE(*,2)'  EFFECTIVELY-ZERO VELOCITY'
      READ(*,FMAT)X(16)
      OPEN(3,FILE=FINN,STATUS='NEW',FORM='FORMATTED')
      WRITE(3,11)FINN
      DO 2800 I=1,16
```

```
2800     WRITE(3,FMAT)X(I)
         CLOSE(3)
        ENDIF

        END


        SUBROUTINE  INFORM
!       3 SEP 1984
        DIMENSION LINE(25)
        CHARACTER*72 LINE
        CHARACTER*15 FINN,FNAME
1       FORMAT(I3)
2       FORMAT(A)
3       FORMAT(4X,A)
4       FORMAT(A,\)
        FINN='OCDDATA.FAX'
        L=0
        OPEN(1,FILE=FINN,FORM='FORMATTED')
        READ(1,2)FNAME
        READ(1,1)I
        IF  (I.NE.678) GO TO 9
C       I: NUMBER OF LINES OF TEXT UNIT TO BE DISPLAYED
6       READ(1,1)I
        IF  (I.EQ.0) GO TO 9
        N=0
        IF  (I.LE.20) N=(24-I)/2
        CALL CLEAR(25)
        DO 7 K=1,I
7       READ(1,2)LINE(K)
        DO 8 K=1,I
8       WRITE(*,3)LINE(K)
        IF (N.GT.0) CALL CLEAR(N)
        PAUSE
        IF  (L.EQ.1 ) GOTO 6
        WRITE(*,4)' DISPLAY PROGRAM NOTES ? [ 0/1 = N/Y ]     '
        READ(*,1)L
        IF  (L.EQ.1) GOTO 6
9       CLOSE(1)
        CALL CLEAR(25)
        END


        SUBROUTINE CLEAR(J)
1       FORMAT(A)
        DO 2 I=1,J
2       WRITE(*,1)' '
        END
```

Appendix B

Glossary

GLOSSARY OF PROGRAMS 'FINDCD' & 'FROUDE'

NAME	REFERENCE	DIMENSION	DESCRIPTION
A(N)	constant	-----	RUNGE-KUTTA AND OTHER CONSTANTS
ABL	UPSON	-----	BALLOON SURFACE AREA / GL**2
AC3	default	-----	CONVERGENCE ROUTINE PARAMETER (0 or 1)
AC4	default	-----	CONVERGENCE ROUTINE PARAMETER (0 or 1)
AD	input	ft**2	DUCT AREA, TOTAL
AG	Model	-----	EFFECTIVE UV ABSORPTANCE OF GAS
AGAS	input	-----	IR RESPONSE CONTROL, GAS [0 or 1]
ALF	def/inp	lbm/ft/sec	VISCOSITY COEFFICIENT, GAS
ALIM	input	ft	ALTITUDE GROWTH LIMIT IN TIME DT
AM	Model	mass units	OPTICAL AIR MASS
AR		-----	EFFECTIVE IR ABSORPTANCE, FILM (= ER)
ARMS	Model	mass units	OPTICAL AIR MASS
ASTOP	input	ft	ALTITUDE, COMPUTATION ABORT (see notes)
AV	Model	-----	EFFECTIVE UV ABSORPTANCE OF FILM
AYR	def/inp	-----	COEFF. OF ABSORPTIVITY, IR
AYRG	def/inp	-----	COEFF. OF ABSORPTIVITY, SOLAR (gas)
AYRG1	formula	-----	(1-AYRG)
AYV	def/inp	-----	COEFF. OF ABSORPTIVITY, SOLAR
B(N)	constant	-----	RUNGE-KUTTA AND OTHER CONSTANTS
BE	formula	lb/ft**3	SPECIFIC LIFT
BET	def/inp	-----	VISCOSITY EXPONENT, GAS
BP	VIRON	mb	PRESSURE, ATMOSPHERIC (not used)
BUOY		lb	BUOYANCY, TOTAL
BZ	constant	lb/ft/sec/R**4	STEFAN-BOLTZMAN CONST. (3.6995E-10)
C(N).	constant	-----	RUNGE-KUTTA AND OTHER CONSTANTS
CA	VIRON	lb/sec/R	COEFFICIENT, THERMAL CONDUCTIVITY OF AIR
CC	constant	GKm/Kelvin	VALUE (34.163195)
CD		-----	COEFFICIENT, DRAG
CF	def/inp	ft/Rankine	SPECIFIC HEAT OF BALLOON FILM
CG	formula	lb/sec/R	COEFFICIENT, THERMAL CONDUCTIVITY OF GAS
CLEN	input	ft	LENGTH, CAP
CM	def/inp	-----	COEFFICIENT, VIRTUAL MASS
CQ1	def/inp	-----	CORRECTION FACTOR FOR Q1
CQ10	def/inp	-----	CORRECTION FACTOR FOR Q10
CQ11	def/inp	-----	CORRECTION FACTOR FOR Q11
CQ2	def/inp	-----	CORRECTION FACTOR FOR Q2
CQ3	def/inp	-----	CORRECTION FACTOR FOR Q3
CQ4	def/inp	-----	CORRECTION FACTOR FOR Q4
CQ5	def/inp	-----	CORRECTION FACTOR FOR Q5
CQ6	def/inp	-----	CORRECTION FACTOR FOR Q6
CQ7	def/inp	-----	CORRECTION FACTOR FOR Q7
CQ8	def/inp	-----	CORRECTION FACTOR FOR Q8
CQ9	def/inp	-----	CORRECTION FACTOR FOR Q9
CS	default	-----	SOLAR RADIATION FACTOR
CV	def/inp	ft/Rankine	SP. HEAT AT CONST. VOL., GAS
CX	formula	-----	COS(XL)*COS(XD)
D(M,N)	formulas	-----	TIME DERIVATIVE OF Y(M,N)
DAL		ft	ALLOWABLE ALTITUDE CLOSURE

DB	default	lb/sec	ACTUAL DEBALLASTING RATE (default is 0)
DBB		lb/sec	MAXIMUM DEBALLASTING RATE (converted)
DBB	input	lb/min	MAXIMUM DEBALLASTING RATE
DCD	input	-----	CD ADJUSTMENT
DD0	VIRON	GKm	STORAGE [EF] for altitude ZZ
DD1	VIRON	GKm	STORAGE [EF]
DD2	VIRON	Kelvin	STORAGE [EG] for altitude ZZ
DD3	VIRON	Kelvin	STORAGE [EG]
DD4	VIRON	mb	STORAGE [PX] for altitude ZZ
DD5	VIRON	mb	STORAGE [PX]
DD6	VIRON	Kelvin/GKm	STORAGE [RR] for altitude ZZ
DD7	VIRON	Kelvin/GKm	STORAGE [RR]
DD8	VIRON	Kelvin	STORAGE [DY] for altitude ZZ
DD9	VIRON	Kelvin	STORAGE [DY]
DM	MYBLN	ft	DIAMETER OF BALLOON
DRAG	formula	lb	DRAG, AERODYNAMIC
DT	default	sec	TIME, INTEGRATION INCREMENT
DTI	Model	Rankine/GKm	TEMPERATURE GRADIENT, IR
DTM	def/inp	sec	TIME, MAX. INTEGRATION INCREMENT (20.)
DTV	def/inp	sec	TIME, VENTING INTEGRATION INCREMENT (0.
DTX		sec	INTERMEDIATE VALUE OF DT
DU1	input	-----	DUMMY (unused)
DU2	input	-----	DUMMY (unused)
DU3	input	-----	DUMMY (unused)
DU4	input	-----	DUMMY (unused)
DU5	input	-----	DUMMY (unused)
E(N,1)	input	GKm	ALTITUDE
E(N,2)	input	Kelvin	TEMPERATURE, ATMOSPHERIC PROFILE
EG	Model	-----	EFFECTIVE IR EMISSIVITY OF GAS
EGAS		-----	IR RESPONSE CONTROL, GAS [= AGAS]
EI	Model	-----	RADIATIVE EXCHANGE COEFF., GAS & WALL
ELL	input	ft	ELEVATION, LAUNCH SITE
ER	Model	-----	EFFECTIVE IR EMISSIVITY OF WALL
ERR	input	-----	ALLOWABLE R.E. IN COMPUTED ALTITUDE
EYR	def/inp	-----	COEFF. OF EMISSIVITY, IR
EYRG	Model	-----	COEFF. OF EMISSIVITY (gas)
EYRG1	formula	-----	(1 - EYRG)
F(N)	formulas	(variable)	(locally defined)
FF	Model	-----	DIRECTIONAL HEMISPHERICAL REFLECTIVITY
FL	input	-----	FREE LIFT, FRACTIONAL
FLY(N,1)	input	sec	TIME
FLY(N,2)	input	ft	ALTITUDE
FORCE		lb	LIFT, NET
FR		-----	FROUDE NUMBER
FRD		-----	FROUDE NUMBER
FR(N,1)	output	-----	REYNOLDS NUMBER
FR(N,2)	output	-----	FROUDE NUMBER
FR(N,3)	output	-----	FRACTIONAL VOLUME
FR(N,4)	output	-----	DRAG COEFFICIENT
FV	Model	lb/ft/sec	UV FLUX
G	constant	ft/sec**2	GRAVITATIONAL CONSTANT (32.1741)
GAM	def/inp	ft lb/lbm/R	THERMAL CONDUCTIVITY COEFF., GAS
GB	constant	-----	(CLEN/GL): used once as DUMMY
GB	MYBLN	-----	(GN/GL)
GG1	input	(variable)	DUMMY (locally defined)

GG2	input	(variable)	DUMMY (locally defined)
GH	input	degrees	GREENWICH HOUR ANGLE
GL	input	ft	GORELENGTH, BALLOON
GLN10	constant	------	NATURAL LOG OF 10
GN	MYBLN	ft	GORELENGTH, GENERATOR SHAPE
GN1	def/inp	-----	CORRECTION FACTOR
GN2	def/inp	------	CORRECTION FACTOR
GN3	def/inp	-----	CORRECTION FACTOR
GNU	Model	-----	NUSSELT NUMBER
GP	PRGR	-----	GRASHOF NO. * PRANDTL NO.
GS	constant	lb/ft/sec	SOLAR CONSTANT (96)
HC	MYBLN	ft**2	AREA,UV ABSORPTION
IN		------	INT(XIN)
INN		------	INT(XINN), subject to option
INN	input	------	NO. OF FLIGHT DATA POINTS TO BE ANALYZED
IVENT		------	STATUS OF DUCT VENTING (Y/N = 1/0)
JSEG	input	-----	NUMBER OF DRAG MODEL SEGMENTS
KI		------	INDEX, RUNGE-KUTTA
KJ		------	INDEX, RUNGE-KUTTA
KK		------	INDEX, RUNGE-KUTTA
LAUNCH	default	------	STATUS OF LAUNCH (Y/N = 1/0)
LEAP	default	------	STATUS (TT.GT.DT / TT.LE.DT = 0 / 1)
LL0	VIRON	------	STORAGE [JX] for altitude ZZ
LL1	VIRON	------	STORAGE [JX]
LL2	VIRON	------	STORAGE [JV] for altitude ZZ
LL3	VIRON	------	STORAGE [JV]
P0	input	mb	PRESSURE, ATMOSPHERIC AT LAUNCH SITE
PAM	VIRON	mb	PRESSURE, ATMOSPHERIC (at alt. ZZ)
PE	VIRON	lb/ft**2	PRESSURE, ATMOSPHERIC
PG	formula	lb/ft**2	PRESSURE DIFFERENTIAL, BALLOON APEX
Q(M,N)	formulas	(variable)	VARIABLES, RUNGE-KUTTA
Q1	Model	ft lb/sec	CONVECTIVE HEAT TRANSFER, GAS & WALL
Q10	Model	ft lb/sec	INFRARED EMISSION, GAS
Q11	Model	ft lb/sec	INFRARED ABSORPTION, GAS
Q2	Model	ft lb/sec	DIRECT SOLAR ENERGY ABSORPTION
Q3	Model	ft lb/sec	INFRARED ABSORPTION
Q4	Model	ft lb/sec	CONVECTIVE HEAT TRANSFER, WALL & AIR
Q5	Model	ft lb/sec	INFRARED EMISSION
Q6	Model	ft lb/sec	REFLECTED SOLAR ENERGY ABSORPTION
Q7	Model	ft lb/sec	RADIATIVE EXCHANGE, GAS & WALL
Q8	Model	ft lb/sec	DIRECT SOLAR ENERGY ABSORPTION, GAS
Q9	Model	ft lb/sec	REFLECTED SOLAR ENERGY ABSORPTION, GAS
QA	formula	------	COS(SOLAR ZENITH ANGLE)
QB	formula	------	COS(fcn[altitude])
QQ	formula	(variable)	DUMMY, RUNGE-KUTTA
RA	constant	ft/Rankine	GAS CONSTANT, AIR (53.352)
RE	constant	ft	RADIUS OF EARTH (20,855,278)
REV(M,N)	input	------	EXPONENTS AND COEFFICIENTS, DRAG MODEL
RG	constant	ft/Rankine	GAS CONSTANT, HELIUM (386.076)
RL	Model	------	REFLECTANCE
RM	MYBLN	ft	RADIUS, GAS BUBBLE
RN	formula	------	REYNOLDS NUMBER
RO	VIRON	slug/ft**3	DENSITY, AIR
RYR	def/inp	------	COEFF. OF REFLECTIVITY, IR
RYV	def/inp	------	COEFF. OF REFLECTIVITY, SOLAR

SA	Model	ft**2	EFFECTIVE SURFACE AREA OF GAS BUBBLE
SB	MYBLN	-----	AREA OF BALLOON BUBBLE, FRACTIONAL
SIG	input	-----	SIGMA, BALLOON SHAPE FACTOR
SIGX	UPSON	-----	SIGMA, IN FLIGHT
SPD	input	knots	LAUNCH WIND SPEED (converted to ft/sec)
SPDSQ		(ft/sec)**2	(not used)
SPEED		ft/sec	ASCENT RATE (or launch wind speed)
SR	Model	ft**2	EFFECTIVE IR SURFACE AREA
SW	VIRON	lb/ft**3	SPECIFIC WEIGHT, AIR
SWG	formula	lb/ft**3	SPECIFIC WEIGHT, GAS
SX	formula	-----	SIN(XL) * SIN(XD)
T		sec	TIME, TOTAL ELAPSED
TCP	input	mils	THICKNESS OF CAPS (sum)
THP	input	mils	THICKNESS, BALLOON WALL
TI	VIRON	Rankine	TEMPERATURE, EQUILIBRIUM RADIATION
TIR1	Model	Rankine	TEMPERATURE, IR (at tropopause)
TIRO	Model	Rankine	TEMPERATURE, IR (at msl)
TK	VIRON	Kelvin	TEMPERATURE, ATMOSPHERIC
TLIM	input	Rankine	TEMPERATURE GROWTH LIMIT
TR	VIRON	Rankine	TEMPERATURE, ATMOSPHERIC
TRM1	Model		TRANSMITTANCE, ATMOSPHERIC (solar)
TRM2	Model		TRANSMITTANCE, ATMOSPHERIC (solar)
TROP	Model	GKm	HEIGHT, TROPOPAUSE
TS	input	sec	TIME, TEMPERATURE STABILIZATION
TSTOP	input	sec	TIME, COMPUTATION ABORT (see notes)
TT		sec	TIME, ELAPSED PRIOR TO STABILIZATION
TT		sec	TIME, REMAINING IN INTERVAL
TTT		sec	STORAGE [T]
TX	VIRON	Kelvin	TEMPERATURE AT ALTITUDE ZZ (not used)
TYR	def/inp	-----	COEFF. OF TRANSMISSIVITY, IR
TYV	def/inp	-----	COEFF. OF TRANSMISSIVITY, SOLAR
U	NELSON	-----	FRACTIONAL PART OF UO
UO	NELSON	-----	RATIO: CAP AREA / BALLOON SURFACE AREA
VA		ft**3	VOLUME OF GAS, PRELAUNCH (no superheat)
VB		-----	VOLUME OF BALLOON, FRACTIONAL
VD	formula	ft**3/sec	VOLUME FLOW RATE THROUGH DUCTS
VL	def/inp	ft/sec	EFFECTIVE ZERO VELOCITY
VLIM	input	ft/sec	VELOCITY GROWTH LIMIT
VS	VIRON	lb/ft/sec	VISCOSITY, AIR
VSG	formula	lb/ft/sec	VISCOSITY, GAS
VT		ft**3	VOLUME, AT CEILING ALTITUDE
VTM	input	ft**3	VOLUME OF BALLOON, THEORETICAL MAXIMUM
VU	UPSON	-----	RATIO: INSTANTANEOUS VOLUME / GL**3
VV	Model	-----	DISCHARGE COEFFICIENT, EV-13
VV	formula	ft**3/sec	VOLUME FLOW RATE THROUGH EV-13
W(M,N)	NELSON	-----	COEFFICIENTS, CAP AREA
WB	input	lb	WEIGHT, BALLOON
WC	formula	lb	EFFECTIVE WEIGHT, CAPS
WE	formula	lb	EFFECTIVE WEIGHT, BUBBLE ENVELOPE
WOW	def/inp	lb/ft**2/mil	UNIT WEIGHT (POLYETHYLENE = 0.0048)
WP	input	lb	WEIGHT, IRREDUCIBLE PAYLOAD
WS		lb	WEIGHT, SYSTEM
WT	input	lb	WEIGHT, BALLAST
WTX		lb	STORAGE [WT]
WV		sec/ft**2	STORAGE [SW/VS]

X		(variable)	DUMMY (locally defined)
XD	input	degrees	DECLINATION, LAUNCH
XG	input	degrees	LONGITUDE, LAUNCH
XIN	input	-----	NO. OF LOCAL ATMOSPHERE POINTS (MAX=30)
XINN	input	-----	NO. OF FLIGHT DATA POINTS (MAX=100)
XL	input	degrees	LATITUDE, LAUNCH
XXAL		ft	STORAGE [Y(3,5)]
XXHC		ft**2	STORAGE [HC]
XXRM		ft	STORAGE [RM]
XXVB		-----	STORAGE [VB]
XXWV		sec/ft**2	STORAGE [WV]
Y		(variable)	DUMMY (locally defined)
Y(1,N)	formulas	ft/sec	VELOCITY, ASCENT
Y(2,N)	formula	lb	WEIGHT, PAYLOAD
Y(3,N)	formula	ft	ALTITUDE
Y(4,N)	formula	Rankine	TEMPERATURE, FILM
Y(5,N)	formulas	lb	WEIGHT, GAS
Y(6,N)	formulas	Rankine	TEMPERATURE, GAS
Y(7,N)	formulas	ft**3	VOLUME, GAS
Y(8,N)	formulas	lb/ft**2	PRESSURE, GAS
ZFIT		ft	ALTITUDE CLOSURE
ZS	Model	degrees	SOLAR ZENITH ANGLE
ZZ	Model	ft	ALTITUDE, OPTICAL AIR MASS

NOTES: 'def/inp' indicates default value can be changed interactively.
References to 'Model','formula' and 'formulas' can be found in program 'FROUDE' listing.
Required definitions of subroutine terms can also be found in program 'FROUDE' listing.
Terms 'ASTOP' and 'TSTOP' used in program 'FROUDE' have no default values; select reasonable values based on expected flight profile.